

8-11-1997

Evaluation of two applications of spectral mixing models to image fusion

Gary Robinson

Follow this and additional works at: <http://scholarworks.rit.edu/theses>

Recommended Citation

Robinson, Gary, "Evaluation of two applications of spectral mixing models to image fusion" (1997). Thesis. Rochester Institute of Technology. Accessed from

This Thesis is brought to you for free and open access by the Thesis/Dissertation Collections at RIT Scholar Works. It has been accepted for inclusion in Theses by an authorized administrator of RIT Scholar Works. For more information, please contact ritscholarworks@rit.edu.

EVALUATION OF TWO APPLICATIONS OF SPECTRAL MIXING MODELS TO IMAGE FUSION

by

Gary D. Robinson

Captain, USAF

A thesis submitted in partial fulfillment of the
requirements for the degree of Master of Science
in the Center for Imaging Science,
Rochester Institute of Technology

August 1997

Signature of

Author _____ Gary D. Robinson

Accepted

by _____ Henry E. Rhody 8/11/97

Coordinator, M.S. Degree Program

Date

CHESTER F. CARLSON
CENTER FOR IMAGING SCIENCE
COLLEGE OF SCIENCE
ROCHESTER INSTITUTE OF TECHNOLOGY
ROCHESTER, NEW YORK

CERTIFICATE OF APPROVAL

M.S. DEGREE THESIS

The M.S. Degree Thesis of Gary D. Robinson
has been examined and approved by the
thesis committee as satisfactory for the
thesis requirement for the Master of Science Degree

Dr. John R Schott, Thesis Advisor

Dr. Robert Fiete

Rolando Raqueno

8/11/97

Date

THESIS RELEASE PERMISSION
ROCHESTER INSTITUTE OF TECHNOLOGY
COLLEGE OF SCIENCE
CHESTER F. CARLSON CENTER FOR IMAGING SCIENCE

Title of Thesis: Evaluation of Two Applications of Spectral Mixing Models to Image Fusion

I Gary D. Robinson, hereby grant permission to the Wallace Memorial Library of R.I.T. to reproduce my thesis in whole or in part. Any reproduction will not be for commercial use or profit.

Signature: _____

Date: _____ 8/11/97

Acknowledgments

I want to thank all the members of my thesis committee for the help and guidance they provided during the course of this research. A specific acknowledgment goes to my thesis advisor, Dr. John Schott for helping to answer questions and for providing ideas for different avenues of investigation.

I would also like to thank the DIRS staff and students. The staff kept the computers running and provided me with valued assistance with DIRSIG. My fellow students took time to listen to my research difficulties and offer suggestions, many of which provided successful alternatives for my research.

Finally, I thank my wife, Kristi for her unending love and support.

Table of Contents

1. INTRODUCTION	1
1.1 SPATIAL VS SPECTRAL RESOLUTION.....	1
1.2 CORRELATION.....	2
1.3 MIXED PIXELS.....	3
1.4 SCALE FACTOR.....	4
1.5 OBTAINING HIGH RESOLUTION MATERIAL MAPS	5
1.6 OUTLINE.....	7
2. BACKGROUND AND LITERATURE REVIEW.....	9
2.1 EXISTING IMAGE FUSION METHODS	9
2.1.1 <i>Image Merging for Enhancement of Visual Display</i>	10
2.1.2 <i>Image Merging by Separate Manipulation of Spatial Information</i>	11
2.1.3 <i>Image Merging Which Maintains Radiometric Fidelity</i>	12
2.2 SPECTRAL UNMIXING METHODS	19
2.2.1 <i>Tricorder</i>	20
2.2.2 <i>Spectral Mixture Analysis (Traditional Unmixing)</i>	23
2.2.3 <i>Constraint Conditions</i>	27
2.3 IMAGE FUSION VIA STEPWISE UNMIXING AND SHARPENING	29
2.3.1 <i>Stepwise Unmixing</i>	30
2.3.2 <i>Constraints</i>	33
2.3.3 <i>Sharpening</i>	35
3. APPROACH	39
3.1 USE OF SYNTHETIC IMAGERY (SIG)	39
3.2 TEST METHOD OVERVIEW.....	40
3.3 SELECTION OF TEST IMAGES	42
3.4 CREATION OF PANCHROMATIC DATA SETS	50
3.5 GENERATING FRACTION MAPS	51
3.6 ENDMEMBER SELECTION.....	52
3.6.1 <i>Maximum Noise Fraction (MNF) Transform</i>	53
3.6.2 <i>Pixel Purity Index</i>	54
3.7 OBTAINING SHARPENING LIBRARY	61
3.8 ERROR METRICS.....	63
3.8.1 <i>Squared Error</i>	63
3.8.2 <i>Effective RMS</i>	64
3.8.3 <i>Effective Edge RMS</i>	64
3.9 GENERATING OUTPUT IMAGES	65
3.9.1 <i>Fusion</i>	65
3.9.2 <i>Unmixing</i>	68
3.9.3 <i>Sharpening</i>	76
4. RESULTS.....	82
4.1 PC vs UNIX	82
4.2 CPU TIME	82
4.3 RMS ERROR	83

4.4 EFFECTS OF SCALE	86
4.5 NUMBER OF ENDMEMBERS	89
4.6 EFFECTS OF SHADOW	91
4.7 VISUAL EVALUATION	103
5. CONCLUSIONS AND RECOMMENDATIONS	113
5.1 CONCLUSIONS BASED ON QUANTITATIVE DATA (TRUTH)	113
5.2 CONCLUSIONS BASED ON QUALITATIVE DATA (VISUAL)	114
5.3 PROPOSED REVISION TO SQUARED ERROR METRIC	114
5.4 RECOMMENDATIONS	115
5.4.1 <i>Algorithm Improvements</i>	116
APPENDIX A: SOLVING THE LSE PROBLEM	120
APPENDIX B: SOLVING THE LSI PROBLEM	122
APPENDIX C: DATA SETS	126
6. REFERENCES	131

LIST OF FIGURES

FIGURE 1: SPECTRAL BANDPASSES OF TM AND SPOT PAN BANDS.....	2
FIGURE 2: BASIC MIXTURE TYPES	4
FIGURE 3: ILLUSTRATION OF SUPERPIXELS AND SUBPIXELS	5
FIGURE 4: UNMIX AND SHARPEN IMAGE FUSION PROCESS.....	6
FIGURE 5: SHARPEN AND UNMIX IMAGE FUSION PROCESS.....	7
FIGURE 6: EXAMPLE LOOK-UP TABLE	15
FIGURE 7: POSSIBLE SUPERPIXEL NEIGHBORHOODS IN EXTENDED REGRESSION METHOD.....	17
FIGURE 8: SAMPLE SPECTRA USING TRICORDER ALGORITHM (CLARK <i>ET AL</i> , 1990)	21
FIGURE 9: THREE MATERIAL MIXTURES IN TWO SPECTRAL BANDS	28
FIGURE 10: MIXTURE REQUIRING NEGATIVE FRACTIONS.....	28
FIGURE 11: ILLUSTRATION OF SHARPENING	35
FIGURE 12: TEST PLAN OVERVIEW	41
FIGURE 13: FOREST TEST SCENE (COLOR IMAGE).....	44
FIGURE 14: ROCHESTER DIRSIG SCENE (COLOR IMAGE)	46
FIGURE 15: BAND 4 (570 - 650 nm) OF DAEDALUS IMAGE	47
FIGURE 16: BAND 15 (749 - 760 nm) OF HYDICE IMAGE.....	48
FIGURE 17: PERFECTLY UNMIXED MATERIAL MAPS FOR GRASS, DIRT, & DECIDUOUS.....	52
FIGURE 18: PERFECTLY UNMIXED MATERIAL MAPS FOR GRASS, WATER, ROOF GRAVEL, LOAM, TREES,	52
FIGURE 19: LOCATING EXTREMA IN THE PIXEL PURITY INDEX	54
FIGURE 20: OBTAINING ENDMEMBERS FROM PPI CLUSTERS	55
FIGURE 21: FOREST SPECTRAL CURVES	57
FIGURE 22: ROCHESTER SPECTRAL CURVES	58
FIGURE 23: DADEALUS SPECTRAL CURVES.....	59
FIGURE 24: HYDICE SPECTRAL CURVES	61
FIGURE 25: SPECTRAL CURVES FOR SHARPENING BANDS (HYDICE).....	62
FIGURE 26: COMPARISON OF OUTPUT FROM SIMPLE RATIO AND GLOBAL REGRESSION METHODS	66
FIGURE 27: FUSION PROGRAM FLOW DIAGRAM.....	67
FIGURE 28: SAMPLE FUSION HEADER	68
FIGURE 29: STEPWISE UNMIXING PROGRAM FLOW DIAGRAM.....	70
FIGURE 30: TRADITIONAL UNMIXING PROGRAM FLOW DIAGRAM.....	71
FIGURE 31: SAMPLE UNMIXING HEADER.....	72
FIGURE 32: SHARPENING PROGRAM FLOW DIAGRAM	77
FIGURE 33: SAMPLE SHARPENING HEADER	78
FIGURE 34: SQUARED ERROR AND RUN TIMES (FOREST SCENE)	83
FIGURE 35: RMS ERRORS FOR FUSION OF FOREST IMAGE	84
FIGURE 36: RMS ERRORS FOR FUSION OF DAEDALUS IMAGE.....	85
FIGURE 37: RMS ERRORS FOR FUSION OF ROCHESTER IMAGE	86
FIGURE 38: UNMIXING FOREST SCENE AT VARIOUS RESOLUTIONS.....	87
FIGURE 39: UNMIXING ROCHESTER SCENE AT VARIOUS RESOLUTIONS	87
FIGURE 40: IMAGE ENHANCEMENT FOR FOREST SCENE AT VARIOUS SCALE FACTORS.....	88
FIGURE 41: IMAGE ENHANCEMENT FOR ROCHESTER SCENE AT VARIOUS SCALE FACTORS	89
FIGURE 42: RESULTS OF TRADITIONAL UNMIXING FOR ROCHESTER SCENE WITH VARIOUS NUMBERS OF ENDMEMBERS	90
FIGURE 43: EFFECT OF SHADOW ENDMEMBER (STEPWISE UNMIX/SHARPEN).....	92

FIGURE 44: EFFECT OF SHADOW ENDMEMBER (FUSE/UNMIX).....	93
FIGURE 45: EFFECT OF SHADOW ENDMEMBER (TRADITIONAL UNMIX/SHARPEN).....	93
FIGURE 46: KEY TO FOREST SCENE FRACTION MAPS.....	94
FIGURE 47: FOREST SCENE FRACTION MAPS (STEPWISE UNMIX/SHARPEN) WITHOUT SHADOW ENDMEMBER ...	95
FIGURE 48: FOREST SCENE FRACTION MAPS (STEPWISE UNMIX/SHARPEN) WITH SHADOW ENDMEMBER	96
FIGURE 49: FOREST SCENE FRACTION MAPS (TRADITIONAL UNMIX/SHARPEN) WITHOUT	97
FIGURE 50: FOREST SCENE FRACTION MAPS (TRADITIONAL UNMIX/SHARPEN) WITH.....	98
FIGURE 51: FOREST SCENE FRACTION MAPS (FUSE/UNMIX METHOD) WITHOUT SHADOW ENDMEMBER	99
FIGURE 52: FOREST SCENE FRACTION MAPS (FUSE/UNMIX METHOD) WITH SHADOW ENDMEMBER.....	100
FIGURE 53: FOREST SCENE TRUTH FRACTION MAPS (4X)	101
FIGURE 54: ERROR CALCULATED FROM SHADOW CLASS MAP VS ERROR CALCULATED FROM CLASS MAP WITHOUT SHADOW (ONE OF TWO).....	102
FIGURE 55: ERROR CALCULATED FROM SHADOW CLASS MAP VS ERROR CALCULATED FROM CLASS MAP WITHOUT SHADOW (TWO OF TWO)	103
FIGURE 56: KEY TO DAEDALUS SCENE FRACTION MAPS	104
FIGURE 57: FRACTION MAPS FOR DAEDALUS IMAGE (STEPWISE UNMIX/SHARPEN)	104
FIGURE 58: FRACTION MAPS FOR DAEDALUS IMAGE (TRADITIONAL UNMIX/SHARPEN)	104
FIGURE 59: FRACTION MAPS FOR DAEDALUS IMAGE (FUSE/UNMIX)	105
FIGURE 60: FRACTION MAPS FOR DAEDALUS IMAGE (DEGRADED TO 4X)	105
FIGURE 61: KEY TO ROCHESTER SCENE FRACTION MAPS.....	106
FIGURE 62: FRACTION MAPS FOR ROCHESTER SCENE (FUSE/UNMIX).....	106
FIGURE 63: FRACTION MAPS FOR ROCHESTER SCENE (STEPWISE/SHARPEN).....	107
FIGURE 64: FRACTION MAPS FOR ROCHESTER SCENE (TRADITIONAL/SHARPEN).....	107
FIGURE 65: ROCHESTER SCENE TRUTH FRACTION MAPS (4X)	108
FIGURE 66: KEY TO HYDICE SCENE FRACTION MAPS.....	109
FIGURE 67: FRACTION MAPS FOR HYDICE IMAGE (STEPWISE/SHARPEN) USING ONE SHARPENING BAND	109
FIGURE 68: FRACTION MAPS FOR HYDICE IMAGE (STEPWISE/SHARPEN) USING.....	110
FIGURE 69: FRACTION MAPS FOR HYDICE IMAGE (TRADITIONAL/SHARPEN) USING	110
FIGURE 70: FRACTION MAPS FOR HYDICE IMAGE (TRADITIONAL/SHARPEN) USING	111
FIGURE 71: FRACTION MAPS FOR HYDICE IMAGE (FUSE/UNMIX).....	112
FIGURE 72: FRACTION MAPS FOR HYDICE IMAGE (DEGRADED TO 2X)	112
FIGURE 73: SAMPLE MATERIAL FRACTIONS IN 3X3 PIXEL NEIGHBORHOOD	117

LIST OF TABLES

TABLE 1: BASIC ANOVA TABLE.....	31
TABLE 2: EXTRA SUM OF SQUARES ANOVA TABLE	32
TABLE 3: NECESSARY CONDITIONS TO MINIMIZE L	37
TABLE 4: M-7 (FOREST) SPECTRAL BANDS (μm).....	42
TABLE 5: M-7 (ROCHESTER) SPECTRAL BANDS (μm).....	45
TABLE 6: DAEDALUS SPECTRAL BANDS (μm).....	48
TABLE 7: HYDICE SPECTRAL BANDS (nm).....	49
TABLE 8: FOREST SPECTRAL LIBRARY	56
TABLE 9: ROCHESTER SPECTRAL LIBRARY	57
TABLE 10: DAEDALUS SPECTRAL LIBRARY (μm).....	58
TABLE 11: HYDICE SPECTRAL BANDS (nm).....	61
TABLE 12: SPECTRAL LIBRARY FOR SHARPENING BAND (FOREST).....	62
TABLE 13: SPECTRAL LIBRARY FOR SHARPENING BAND (ROCHESTER)	62
TABLE 14: SPECTRAL LIBRARY FOR SHARPENING BAND (DAEDALUS).....	62
TABLE 15: SPECTRAL LIBRARY FOR SHARPENING BANDS (HYDICE)	63
TABLE 16: DATA FOR FOREST SCENE WITH SHADOW ENDMEMBER (UNCORRECTED FOR SHADOW).....	126
TABLE 17: DATA FOR FOREST SCENE WITH SHADOW ENDMEMBER (CORRECTED FOR SHADOW)	127
TABLE 18: DATA FOR FOREST SCENE WITHOUT SHADOW ENDMEMBER.....	128
TABLE 19: DATA FOR DAEDALUS SCENE.....	129
TABLE 20: DATA FOR ROCHESTER SCENE.....	130

ACRONYMS LIST

ANOVA	-	Analysis of Variance
AVIRIS	-	Airborne Visible/Infrared Imaging Spectrometer
DC	-	Digital Count
DIRS	-	Digital Image Processing and Remote Sensing
DIRSIG	-	Digital Image Processing and Remote Sensing Synthetic Image Generation
ERIM	-	Environmental Research Institute of Michigan
GIFOV	-	Ground Instantaneous Field of View
HPF	-	High Pass Filter
HRP	-	High Resolution Panchromatic
HRXS	-	High Resolution Multispectral
HYDICE	-	Hyperspectral Digital Imagery Collection Experiment
IDL®	-	Interactive Data Language
IHS	-	Intensity Hue Saturation
LDP	-	Least Distance Programming
LRXS	-	Low Resolution Multispectral
LSE	-	Least Squares Equality
LSI	-	Least Squares Inequality
LUT	-	Look-up Table
μm	-	Microns (10^{-6} meter)
MNF	-	Maximum Noise Fraction
MRI	-	Magnetic Resonance Imaging
MS	-	Mean Square
MSE	-	Mean Square (Error/Residual)
MSR	-	Mean Square (Regression)
MSS	-	Multispectral Scanner
NIR	-	Near Infrared (Region of Spectrum)
nm	-	nanometers (10^{-9} meter)
NNLS	-	Non-Negative Least Squares
PC	-	Personal Computer
PCA	-	Principal Components Analysis
PPI	-	Pixel Purity Index
RBV	-	Return Beam Vidicon
RMS	-	Root Mean Square
RSI	-	Research Systems Incorporated
SE	-	Squared Error
SIG	-	Synthetic Image Generation
SIR-A	-	Shuttle Imaging Radar
SPOT	-	Système Pour l'Observation de la Terre
SS	-	Sum of Squares
SSE	-	Sum of Squares (Error/Residual)
SSR	-	Sum of Squares (Regression)
SWIR	-	Short Wave Infrared (Region of Spectrum)
TM	-	Thematic Mapper
VIS	-	Visible (Region of Spectrum)

1. INTRODUCTION

1.1 Spatial vs Spectral Resolution

Remote sensing instruments are capable of obtaining images with high spatial resolution, or high spectral resolution. Spatial resolution refers to how well a sensor can resolve the spatial details of a scene. It is often measured by the sensor's Ground Instantaneous Field of View (GIFOV). The GIFOV is the projection of the detector aperture, through the sensor's optics, onto the ground. A smaller GIFOV refers to a sensor with higher spatial resolution. A small GIFOV can be obtained by using a small detector. However, in order to obtain a sufficient number of photons for useful imaging, and to maintain an adequate signal to noise level, the detector must be sensitive over a relatively wide spectral band. Spectral resolution refers to the width of the bandpass where radiance is measured; the narrower (finer) the spectral resolution, the more bands that can be obtained over a specific spectral range. To obtain high spectral resolution, a narrow filter or grating is added to the detector. In order to obtain sufficient photons the detector must be large, leading to a large GIFOV and low spatial resolution. Two types of remote sensing platforms are commonly used. One type creates high spatial resolution panchromatic images (typically in the visible or near infrared region of the spectrum), and the other type creates multispectral or hyperspectral images with fine spectral resolution.

There will always be some trade-off between spatial and spectral resolution. Images with high spatial resolution can locate objects with high accuracy, whereas images with high spectral resolution can be used to identify materials. With different sensors collecting information over the same area, it is useful to merge the data into a hybrid product containing the useful information of both platforms. Such a hybrid image with high spatial and spectral resolution can be used to create detailed material maps.

1.2 Correlation

Generating hybrid images requires a large amount of correlation in images. Consider the LANDSAT Thematic Mapper (TM) which has six spectral bands in the reflective region ranging from 0.400 μm to 2.350 μm , and the French SPOT panchromatic band which ranges from 0.510 μm to 0.730 μm . As shown in Figure 1, there is spectral overlap between SPOT and TM bands 2 and 3, and the digital counts in the overlap region will be highly correlated. Hybrid images of these bands will show definite, accurate improvements over both original input images. However, fusing SPOT with the infrared bands (e.g. 5 & 7) will be less straightforward. Fusion of these poorly correlated bands requires predictive models to estimate the high-resolution data.

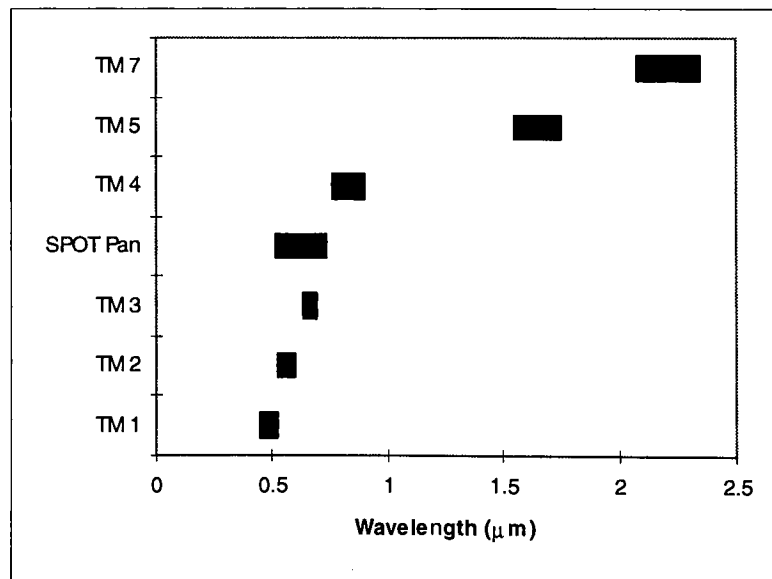


Figure 1: Spectral Bandpasses of TM and SPOT Pan Bands

Most multispectral sensors have bands whose bandpasses range through several regions of the spectrum, including the visible (VIS), near infrared (NIR), and short wave infrared (SWIR). A typical panchromatic sensor will cover a much shorter portion, restriction itself to the VIS or NIR (for example) regions. So image fusion will almost always involve predicting digital counts for poorly correlated bands. The

different methods for performing fusion (discussed in the next chapter) have varying levels of effectiveness. Some level of optimizing to obtain the best estimate will always be involved.

1.3 Mixed Pixels

The region on the ground represented by one pixel in an image may contain a number of materials. The definition of the materials depends on the specific imaging application. For example, if one is looking for broad classifications, pixels may be classified as forest, urban, or water and, except along borders between regions, most pixels can be considered 100% “pure”. However, if the application is more specific, the same pixels can be considered mixtures of deciduous vs coniferous vegetation, or residential vs commercial, or clear vs silty water. So the determination of whether a pixel is mixed or pure often depends upon the specific application.

It is helpful to divide mixtures of materials into three scenarios. Consider first a situation where there are linear interactions between the materials and incident photons. Distinct materials may be mixed at various spatial scales. A mixture is defined as *aggregate* if materials are combined at the macroscopic scale. The total radiance leaving the scene is a spatial average of the individual materials, however, the individual materials cannot be spatially separated by the sensor. An *areal* mixture is also characterized by linear interactions, but involves situations where individual materials can be resolved by the (typically high-resolution) sensor.

The third mixture involves materials combined at the microscopic level. This *intrinsic* mixture involves multiple interactions between materials and incident photons. The average radiance typically depends on a complex combination of the individual material properties. Such mixtures require non-linear models and were not addressed in this work. The three types of mixtures are demonstrated in Figure 2.

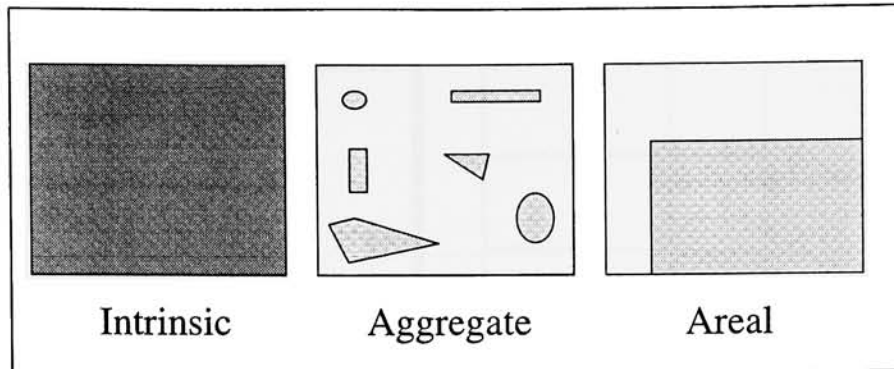


Figure 2: Basic Mixture Types

1.4 Scale Factor

Hybrid images can be produced by fusing low-resolution multispectral images with high-resolution panchromatic images. The pixels of the low-resolution multispectral image (LRXS), often called *superpixels*, cover larger areas of the ground and correspond to several pixels, often called *subpixels*, of the high-resolution panchromatic image (HRP) as illustrated in Figure 3. If the two images have been properly registered, then each LRXS superpixel corresponds to a collection of HRP subpixels equivalent in size to the larger low-resolution pixel.

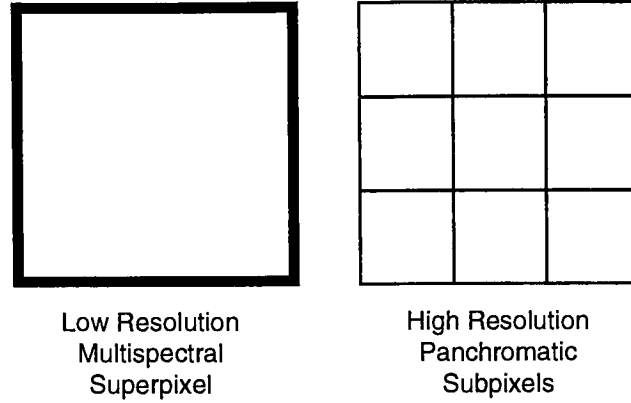


Figure 3: Illustration of Superpixels and Subpixels

The scale factor of the fusion refers to the difference in the GIFOV between the LRXS and HRP images. For example, consider the case where the GIFOV of the LRXS is 30m and that of the HRP is 10m. Then the scale factor is defined to be

$$\text{Scale Factor} = \frac{\text{GIFOV LRXS}}{\text{GIFOV HRP}} = \frac{30}{10} = 3 \quad \text{Eq. 1}$$

Such a fusion scenario will produce a hybrid image with a 3-times (3X) improvement in GIFOV. This hybrid image can then be used to create detailed material maps.

1.5 Obtaining High Resolution Material Maps

There are two steps in creating the detailed material maps previously mentioned. First, the multispectral (or hyperspectral) image is used to identify the materials in the scene. This process, often referred to as *spectral unmixing*, generates several material maps, where each map is an estimate of the percentage of a

specific material within the scene. Second, the material maps and the panchromatic image of the same area serve as constraining inputs to produce *sharpened* material maps, resulting in high-resolution material maps.

One method of image fusion uses the unmix and then sharpen procedure (See Figure 4). An alternate method, theoretically producing identical results, utilizes a sharpen and unmix process. The sharpening produces a high-resolution multispectral image which is then unmixed into high-resolution material maps. There is little published work of image fusion using a sharpen and unmix process (See Figure 5). However, there are several applications which utilize sharpening without further processing (unmixing) of the high-resolution multispectral images.

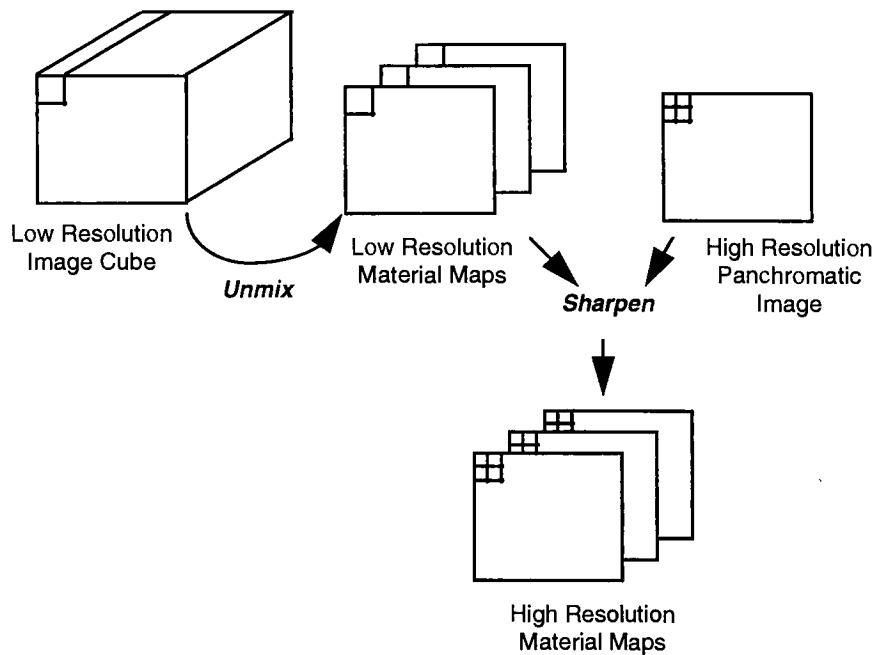


Figure 4: Unmix and Sharpen Image Fusion Process

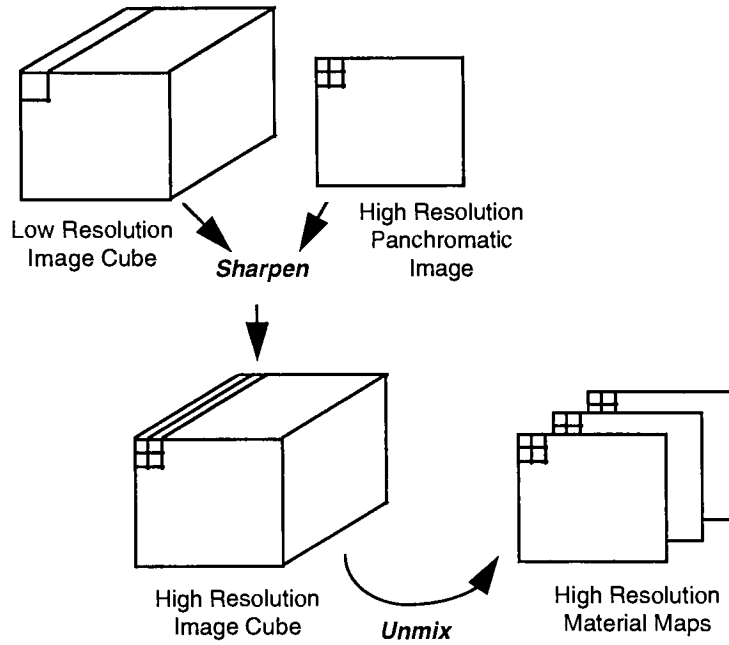


Figure 5: Sharpen and Unmix Image Fusion Process

1.6 Outline

This research implemented image fusion via a sharpen and unmix process and compared the resulting high-resolution material maps to those obtained via an unmix and sharpen process. The unmix and sharpen process employed two methods for producing the low-resolution material maps. A recently developed adaptive unmixing algorithm was compared to traditional unmixing methods. Sharpening was performed on the low-resolution material maps produced by the two unmixing algorithms, and the resulting high-resolution material maps were compared to those generated via the sharpen and unmix process. All methods were evaluated for radiometric fidelity, unmixing accuracy, and enhanced visual display. Quantifiable results are available because synthetic imagery was used in addition to real images.

This document is organized as follows. Section two provides background reference on various image fusion techniques. The specific methods used in this research are discussed in some detail. Section three provides an overview of the test method, including details on steps involved in the image enhancement methods. The quantitative and subjective results of the tests are detailed in section four. The results show that the sharpen/unmix method produces more error than unmixing with the adaptive algorithm and then sharpening. Fraction maps created by the sharpen/unmix method are more visually acceptable, containing more high-frequency information than fraction maps produced by the unmix/sharpen methods. The final section indicates additional avenues of exploration in the area of image fusion.

2. BACKGROUND AND LITERATURE REVIEW

Image fusion involves combining different images into a new hybrid image. The original images may be products of different remote sensing platforms, and may have different spectral and spatial resolutions. For example, we might wish to merge data obtained from the Landsat Thematic Mapper (TM) with that obtained from the French Systeme Pour l'Observation del la Terre (SPOT). The TM has seven spectral bands ranging from .45 to 2.35 microns. Six of the bands (1 - 5 and 7) have 30 meter spatial resolution. The seventh band (band 6) provides thermal information and has 120 meter spatial resolution. SPOT has 3 spectral bands in the visible and near infrared region with 20 meter spatial resolution. It also has a panchromatic band with 10 meter spatial resolution. The most efficient method for an analyst to examine imagery from these two platforms would be to combine the useful information from both into a single image.

Landsat TM and SPOT are not the only types of data that can be merged. Daily *et al.* (1979) and Chavez *et al.* (1983) merged airborne and Shuttle Imaging Radar (SIR-A) images with Landsat Multispectral Scanner (MSS). Lauer and Todd (1981) combined imagery from Landsat MSS with data from the Return Beam Vidicon (RBV). The next generation of hyperspectral space-based sensors is currently in the design phase. These sensors will have high spectral resolution, but very poor spatial resolution. The pending increase in sensors will increase the need for better image fusion applications.

2.1 Existing Image Fusion Methods

There are several existing methods to perform image fusion. Munechika (1990) groups these methods into three classes. The first class is called "Merging Images for Enhancement of Visual Display". These algorithms are primarily concerned with optimizing an image display so that it looks good for the analyst. The second class is called "Image Merging by Separate Manipulation of Spatial Information". These algorithms

merge data by separate manipulation of the spectral and spatial information. The final class is called “Image Merging to Maintain Radiometric Fidelity”. These algorithms merge data, while ensuring that the radiometric accuracy of the original multispectral data is maintained or degraded only minimally.

2.1.1 Image Merging for Enhancement of Visual Display

Image fusion routines that enhance visual display have also been referred to as *ad hoc* methods. The primary concern is to optimize the display for analysis purposes. There is no concern in preserving the radiometric accuracy of the multispectral data. One method used employs histogram specification and contrast stretching. Two examples of generic methods are given by the equations (Welch & Ehlers, 1987)

$$XS'_i = a_i \times \sqrt{XS_i \times P} + b_i \quad \text{Eq. 2}$$

or

$$XS'_i = a_i \times (w_1 XS_i \otimes w_2 P) + b_i \quad \text{Eq. 3}$$

where XS'_i is the digital count (DC) for a pixel in the i^{th} band of the high-resolution hybrid image, XS_i is the digital count for the corresponding pixel in the original multispectral image, P is the digital count for the corresponding pixel in the high-resolution panchromatic image, w_1 and w_2 are weighting factors, and a_i and b_i are scaling factors to optimize the hybrid image for the dynamic range of the display system, and \otimes is an operator which could be addition, subtraction, multiplication, ratio, etc.

A simpler *ad hoc* technique to enhance a RGB display is to replace the green channel with the panchromatic data, leaving the red and blue channels unchanged. Since the human visual system is most sensitive to green, the display looks sharper to the viewer.

2.1.2 Image Merging by Separate Manipulation of Spatial Information

An image can be assumed to contain a low frequency and high frequency component. The low frequency data contains the spectral information, while the high frequency data contains the spatial information. The image fusion algorithms in this class manipulate the spatial (high frequency) component while preserving the spectral component to generate enhanced images. Braun (1992) compared three algorithms of this class.

2.1.2.1 Intensity Hue Saturation (IHS)

The IHS technique (Chavez, 1991) can be applied to three bands of multispectral data. Three multispectral bands are treated as colors (e.g. red, green, blue). The RGB multispectral image is transformed to an intensity, hue, saturation space, where the intensity is assumed to contain most of the spatial information, and the hue and saturation are assumed to contain most of the spectral information. The panchromatic image is then substituted for the intensity of the multispectral image, and an inverse transformation is performed to return the image to a RGB format. The result is a high-resolution image whose spatial content is derived from the panchromatic image, and whose color (spectral) content is derived from the original multispectral data.

This technique is based on the assumption that edge information (essentially the spatial content) is contained within the intensity. The IHS transformation works as long as the panchromatic image is highly correlated with the bands of the multispectral image.

2.1.2.2 Principal Components Analysis (PCA)

The PCA technique (Chavez, 1991) involves calculating the principal component of the multispectral image. This calculation utilizes linear algebra, and transforms a vector of correlated data into orthogonal components. The first principal component contains data common to all the spectral bands, and should be

similar to the panchromatic image. The first principal component image is replaced by the high-resolution panchromatic image. All remaining principal components are assumed to contain the spectral components and are untouched. An inverse principal component is then performed to obtain a new hybrid image.

2.1.2.3 High Pass Filter (HPF)

The high pass filter technique (Schowengerdt, 1980) is based on the theory that an image is composed of a highpass filtered image and a lowpass filtered image. The hybrid image can be constructed by using the high-resolution image to replace the missing edge information in the low-resolution image using the equation

$$HRXS_j = LRXS_j + K_j \cdot H_{PAN} \quad \text{Eq. 4}$$

where $HRXS_j$ is the digital count of a pixel in the j^{th} band of the hybrid multispectral image, $LRXS_j$ is the digital count of the corresponding pixel in the j^{th} band of the low-resolution multispectral image, K_j is a constant designed to control the contrast of the hybrid image, and H_{PAN} is the digital count of the corresponding pixel in the high-resolution band used for the edge details. K_j is chosen appropriately to ensure that the contrast in the hybrid bands is weighted equally by the low-resolution and high-resolution images.

2.1.3 Image Merging Which Maintains Radiometric Fidelity

All of the previously mentioned image fusion methods primarily enhance visual display. The spatial resolution of the hybrid multispectral image improves compared to the original multispectral image. However, the exact radiometric values of the multispectral image are often lost in the process. Any algorithm used to identify materials in a multispectral image relies inherently on the accuracy of the radiometric values within that image. In order to exploit the information in the hybrid images by use of an automated routine, the radiometry of the hybrid image must match as closely as possible the radiometry of the original multispectral image. The three methods discussed in this class are described by Braun (1992).

2.1.3.1 Ratio Methods

The ratio methods are simple image fusion techniques designed to maintain the radiometry of the original image. They require that the panchromatic sharpening image be highly correlated with the multispectral image. The procedure begins by dividing the pixels of the multispectral image into subpixels which are equal in size to the pixels of the high-resolution panchromatic image.

2.1.3.1.1 Pradines' Method

Pradines (1986) uses the following equation to merge SPOT spectral bands with the SPOT panchromatic band:

$$HRXS_i = LRXS_i \cdot \frac{HRP}{\sum_{\text{superpixel}} HRP} \quad \text{Eq. 5}$$

where $HRXS_i$ is the digital count of a subpixel in the high-resolution hybrid image in the i^{th} band, $LRXS_i$ is the digital count of the corresponding subpixel in the i^{th} band of the multispectral image, and HRP is the digital count of the corresponding subpixel in the high-resolution panchromatic image.

2.1.3.1.2 Price's Method

The disadvantage of the Pradines routine is that it does not account for bands that are not highly correlated with the panchromatic image. Price (1987) proposes a two-stage process for dealing with bands that are either weakly or strongly correlated with the panchromatic image. A ratio is used for the strongly correlated bands, which can be written as

$$LRXS_i = a_i \cdot \overline{HRP_s} + b_i \quad \text{Eq. 6}$$

where a_i and b_i are least squares regression coefficients of a linear fit in the i^{th} band, and $\overline{\text{HRP}}_s$ is the digital count of an averaged panchromatic image superpixel. The regression coefficients are found by regressing $\overline{\text{HRP}}_s$ against LRXS_i . The high-resolution multispectral image (HRXS) is obtained by

$$\text{HRXS}'_i = a_i \cdot \text{HRP} + b_i \quad \text{Eq. 7}$$

and

$$\text{HRXS}_i = \frac{\text{LRXS}_i \cdot \text{HRXS}'_i}{\overline{\text{HRXS}}'_{i,s}} \quad \text{Eq. 8}$$

where HRXS'_i is the digital count of the estimate for the i^{th} band of a high-resolution multispectral image, and $\overline{\text{HRXS}}'_{i,s}$ is the average of HRXS'_i over a superpixel.

Braun (1992) reports that stage 1 of the Price routine produces results similar to the Pradines technique. The main difference is that Price uses an estimate for the high-resolution multispectral bands, whereas Pradines simply uses the high-resolution panchromatic image.

Price uses a Look-Up Table (LUT) in stage 2 of his technique for dealing with uncorrelated spectral bands. The LUT is created by first examining the $\overline{\text{HRP}}_s$ values, and recording the corresponding digital count in the low-resolution multispectral image. The mean of these multispectral pixels is calculated and the value is entered into the LUT. Figure 6 shows an example look-up table. The values in the LUT relate the $\overline{\text{HRP}}_s$ digital counts to the multispectral digital counts in the uncorrelated bands. Now the high-resolution estimates are calculated using the LUT values for HRXS' in equation 8.

Average Pan DC ($\overline{HRP_s}$)	DC from Weakly Correlated Multispectral Bands ($\overline{LRXS_j}$)	Mean Low Res Multispectral DC
0	8,8,8,7,9,...	8
1	21,24,22,20,23,...	22
2	17,14,13,16,...	15
...
255

Figure 6: Example Look-Up Table

2.1.3.1.3 DIRS Method (Simple Ratio)

Munichika (1990) presents a routine which is easier to implement than Price's method. This method is designed to provide as much radiometric accuracy as possible, and forms the basis for the Extended Ratio and Global Coefficient methods which will be discussed in later sections. This method is used by the Digital Imagery Processing and Remote Sensing (DIRS) laboratory at RIT and is often referred to as the DIRS Method.

Munichika's method begins by pixel replicating and blurring the high-resolution panchromatic image so that its subpixels are the same size as the pixels of the low-resolution multispectral image. The panchromatic image is registered to the multispectral image to preserve the radiometry of the multispectral image.

The simple ratio method is given by the equation

$$HRXS_i = HRP \cdot \frac{LRXS_i}{\overline{HRP_s}} \quad \text{Eq. 9}$$

This method works well for spectral bands that are highly correlated with the panchromatic image. It can easily be shown that this equation is radiometrically correct by

$$\overline{HRXS_s} = \frac{\sum_{j=1}^N \frac{LRXS_i}{\overline{HRP_s}} HRP_j}{N} = \frac{LRXS_i}{\overline{HRP_s}} \cdot \frac{\sum_{j=1}^N HRP_j}{N} = \frac{LRXS_i}{\overline{HRP_s}} \overline{HRP_s} = LRXS_i \quad \text{Eq. 10}$$

which shows that the average of the digital counts of the hybrid image over a superpixel equals the digital count of the corresponding pixel in the original multispectral image.

Munechika's method does not work well on mixed pixels, so an enhancement is presented in Munechika *et. al.* (1993). For the case of a mixed pixel, the ratio of $LRXS/\overline{HRP}_s$ is not always the best. In this case, a digital count of a panchromatic subpixel is compared to the mean digital counts of neighboring superpixels. If the subpixel's ratio is closer to that of one of the neighboring superpixel values, then that superpixel's mean is used for the $LRXS/\overline{HRP}_s$ ratio in equation 9. This mixed pixel is not necessarily radiometrically accurate on average over a superpixel, but its quantitative performance on a subpixel case exceeds that of the simple ratio method.

2.1.3.1.4 Extended Ratio

The simple ratio method does not maintain radiometric accuracy for weakly correlated bands. The extended ratio method is designed to deal with the case of poor spectral correlation between a given multispectral band and the panchromatic band, and is used in conjunction with the simple ratio method, with the ratio method implemented for correlated bands. A liner relationship is created between the weakly correlated band, the panchromatic band, and any previously predicted band as

$$LRXS_k = a_0 + a_1 \overline{HRP}_s + a_2 LRXS_i + a_3 LRXS_j + ... \quad \text{Eq. 11}$$

where k refers to a weakly correlated multispectral band and i and j are strongly correlated, previously predicted bands. The coefficients a_0 , a_1 , etc. are obtained by performing a regression in a localized neighborhood around the target superpixel using equation 11. See Figure 7 for a diagram of possible superpixel neighborhoods. The regression is first performed using only one strongly correlated/previously predicted band. Additional bands are used if the residuals remain sufficiently large.

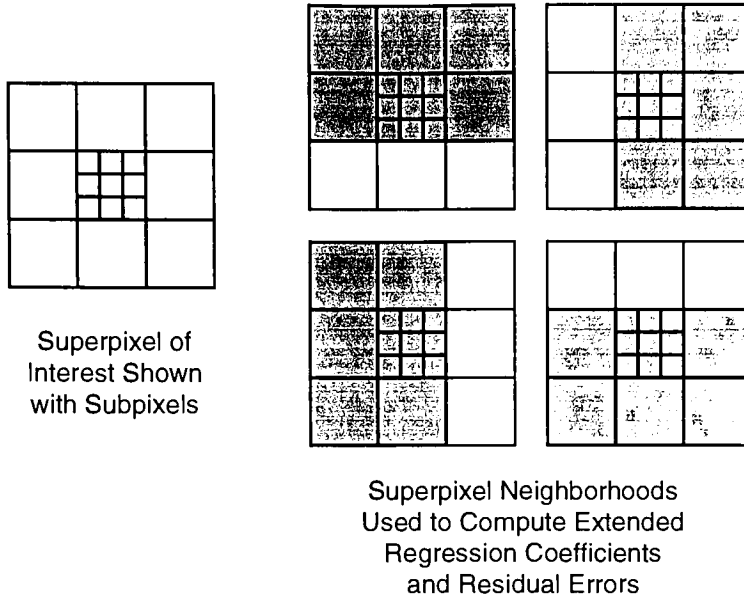


Figure 7: Possible Superpixel Neighborhoods in Extended Regression Method

Once the regression equation is satisfied, the coefficients are used to determine the digital count of the hybrid image subpixels using

$$\text{HRXS}_k = a_0 + a_1 \text{HRP} + a_2 \text{HRXS}_i + a_3 \text{HRXS}_j + \dots \quad \text{Eq. 12}$$

where HRXS_i is the digital count of a hybrid subpixel in band i (previously predicted using the ratio method).

The advantage of the extended regression method is that it allows the hybrid image to be predicted even for poorly correlated bands. In addition, by solving for the coefficients in a localized region around the target superpixel, the extended regression method tends to use superpixels with the same material types as the target superpixel. A problem with the extended regression model is that it produces noisy images when used in areas with uniform digital counts. Any small change in a sharpening band or errors in a previously predicted band are exaggerated when used in estimating hybrid pixel values. However, Braun (1992) notes that extended

regression produces improved results over the simple ratio when used in regions where there is much brightness variation within a material type.

2.1.3.1.5 Global Regression Method

The global regression method is designed to overcome some of the limitations of the extended regression technique. Rather than performing a regression in a localized window around a subpixel, data from the entire image is used. The assumption used is that the best data to solve the regression is from superpixels with the same spectral characteristics as the target subpixel. First, an unsupervised classifier with a large number of classes is used on the multispectral and panchromatic images. A class map is created with all pixels classified into some spectral class (note that no class type needs to be assigned to these classes). The regression in equation 11 is applied using pixels that are in the same class as the target subpixel. The remaining portion of the global regression routine is similar to that for the extended regression technique, with bands incrementally added until the residuals of the regression equation are below a desired threshold. Equation 12 is employed with the coefficients obtained from the regression to obtain the high-resolution multispectral image.

Braun (1992) notes that the global regression technique, on average, outperforms the extended regression routine. The extended regression produces noisy results in low frequency areas, whereas the global regression softens the noise while preserving the edges.

2.1.3.2 Algorithm Summary

Image fusion works best when the low-resolution multispectral image bands are highly correlated with the high-resolution panchromatic band. When there is weak correlation, the quality of the image fusion will be degraded, and routines that separately manipulate spatial data may introduce radiometric inaccuracies. The Intensity Hue Saturation method is in some ways the least robust because it can only be applied to three bands. Braun (1992) states that the routines designed to maintain radiometric accuracy work better than those that

separately manipulate the spatial data. The Price and Munechika methods produce similar results in the correlated bands. The simple ratio technique by Munechika forms the basis for the extended and global regression methods. The extended regression routine works best when scenes contain high frequency information, and the global regression works best when the scene contains medium or low frequency.

2.2 Spectral Unmixing Methods

The multispectral remote sensing platform typically has poor spatial resolution. The large pixel sizes imply that the majority of the pixels in the multispectral image will be mixed. Applications such as mapping vegetation or locating mineral resources require such mixed pixels to be separated into the individual constituents (often called endmembers) whose radiances contribute to the single mixed pixel value. Spectral unmixing transforms the digital counts of mixed pixels into a series of maps which are estimates of the percentage or abundance of the individual materials within a scene.

Spectral unmixing has been used to map many different materials. Images from the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) were used to create individual maps of green vegetation, nonphotosynthetic vegetation, and soil (Roberts *et al* 1993). AVIRIS data was also used to map desert vegetation (Smith 1990). Hyperspectral sensors such as AVIRIS are ideally suited to spectral unmixing applications due to the requirement that there be more spectral bands than constituents to be unmixed.

Spectral unmixing can classify images using scene-derived endmembers or reference endmembers (reflectance spectra measured by field or laboratory instruments). When reference endmembers are used, atmospheric compensation and the responsivity of the sensor must be taken into account.

Two methods of spectral unmixing are prevalent in available research literature. The Spectral Mixture Analysis of Smith *et al* (1990) and Roberts *et al* (1993) provides estimates of the percentages of endmembers employing classical unmixing methods, whereas the Tricorder method of Clark *et al* (1990) produces the abundances by searching for specific absorption features characteristic of individual endmembers.

2.2.1 Tricorder

Clark *et al* (1990) employ a method referred to as Tricorder to determine the individual endmembers within the mixed pixels of a multispectral image. Tricorder is similar to spectroscopic analysis employed by scientists, and employs the same steps used to analyze a spectrum. Endmembers are identified by looking for specific absorption features. For example, kaolinite and dolomite have characteristic absorption features which Tricorder can locate in the spectrum of a mixed pixel.

The following definition of absorption band depth is employed by the algorithm

$$D = 1 - \frac{R_b}{R_c} \quad \text{Eq. 13}$$

where R_b is the reflectance in the center of an absorption feature, and R_c is the reflectance of the continuum at the center of the feature. The continuum is the shape the spectrum would take if the absorption feature were not present. Typically, it is created by simply connecting the wings of the absorption feature with a straight line.

See Figure 8.

The Tricorder algorithm requires that the data be corrected for atmospheric effects. Green *et al* (1993) present a method to calibrate AVIRIS data for atmospheric effects. Assuming the data is corrected for atmosphere, Tricorder uses the following steps. 1) Convolve the library spectra with the sensor response so it resembles the image data. 2) Convert the image data from digital counts to apparent reflectance. 3) Remove the continuum in the library and image spectra using

$$L_c(\lambda) = \frac{L(\lambda)}{C_L(\lambda)} \quad \text{Eq. 14}$$

and

$$S_c(\lambda) = \frac{S(\lambda)}{C_S(\lambda)} \quad \text{Eq. 15}$$

where L_C is the continuum-removed library reflectance spectrum, L is the library spectrum, S_C is the continuum-removed sensor spectrum, S is the sensor spectrum, and C_L and C_S are the continuum spectra estimated from a fit through the wings of the absorption feature.

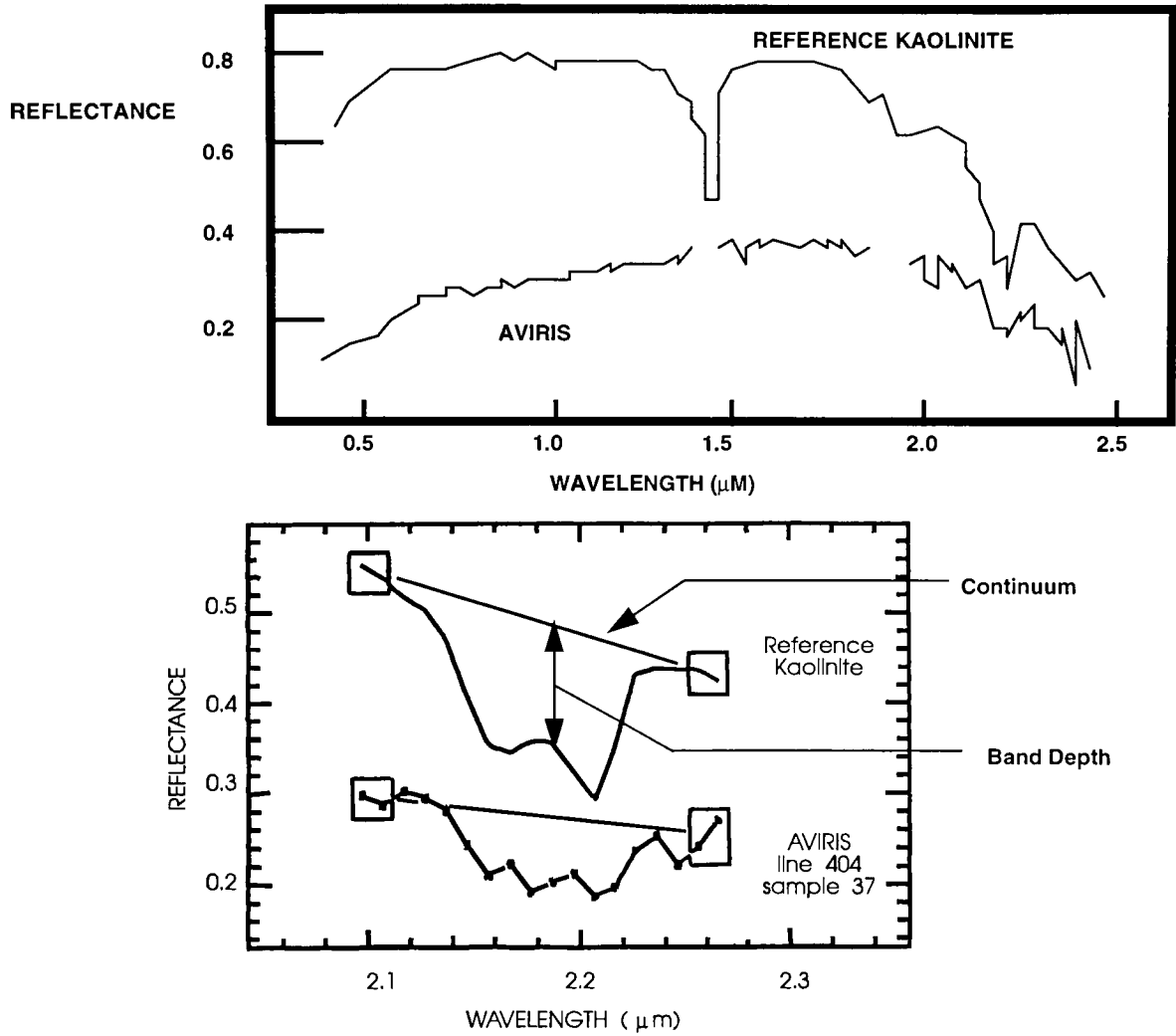


Figure 8: Sample Spectra Using Tricorder Algorithm (Clark *et al*, 1990)

The absorption features in the reference spectrum are typically stronger than in the spectrum recorded by the remote sensor, so the spectral contrast of the reference must be changed to match the contrast of the sensor spectrum. The contrast of the reference spectrum is modified by

$$L'_c = \frac{L_c + k}{1 + k} = a + bL_c \quad \text{Eq. 16}$$

where L'_c is the contrast reduced spectrum that best matches observation, k is a constant, and $a = k/(1+k)$ and $b = 1/(1+k)$.

The coefficients a & b must be determined such that they give the best fit of L'_c to the observed spectrum. A least square fit is performed using

$$a = \frac{\sum S_c - b \sum L_c}{n},$$

$$b = \frac{\sum S_c L_c - \frac{\sum S_c \sum L_c}{n}}{\sum L_c^2 - \frac{(\sum L_c)^2}{n}},$$

and

$$k = \frac{1 - b}{b} \quad \text{Eq. 17}$$

where n is the number of spectral channels in the fit. A material map is produced by fitting a reference spectrum to the spectrum of each pixel in a hyperspectral data set. The band depth is proportional to the abundance of the material and the goodness of fit provides a confidence factor. By plotting $D \cdot R^2$, the bright areas will show high abundance with high confidence in the derived solution. A material is typically characterized by more than one absorption feature, so several features may be used by the algorithm. The features may also be weighted, so that some absorption features take precedence over others.

2.2.1.1 Tricorder Summary

Clark *et al* (1990) use AVIRIS data to demonstrate the Tricorder algorithm, and report good success. This method can even create maps of materials with complicated absorption features (e.g. the kaolinite doublet). Spectral mixture analysis (discussed in the next section) can be accomplished without atmospheric compensation. However, the Tricorder algorithm will not work with data that has not been corrected for atmospheric effects. Another contrast between the two methods is the fact that unmixing with the Tricorder algorithm may only be performed on hyperspectral data, whereas spectral mixture analysis can be performed (with a limited number of endmembers) on multispectral images. Tricorder is also less sensitive to signal to noise in individual channels because many channels are used to map an absorption feature. An inexperienced user cannot simply start working and achieving successful results because expert knowledge of spectroscopy by the user is required for this algorithm. Tricorder was not implemented for this work.

2.2.2 Spectral Mixture Analysis (Traditional Unmixing)

Spectral mixture analysis assumes that for a multispectral image, the spectral variation is due to a small number of endmembers. These endmembers all have different reflectance spectra and the differences in the spectra serve as “fingerprints” to identify the different materials. In the case of areal and aggregate mixtures, it is possible to produce a linear mixture of these endmembers that closely matches the observed spectra measured by the sensor. For N endmembers, this becomes

$$L_{\text{sensor}}(\lambda) = \sum_{e=1}^N L_e(\lambda) f_e \quad \text{Eq. 18}$$

where $L_{\text{sensor}}(\lambda)$ is the spectral radiance reaching the sensor, f_e is the fraction of endmember e within the pixel, and $L_e(\lambda)$ is the spectral radiance of that endmember.

The response of the sensor should be taken into account also. Let

$$DC_i = g_i \cdot L_i + b_i \quad \text{Eq. 19}$$

and

$$L_i = \int_{\lambda} L_{\text{sensor}}(\lambda) \beta_i(\lambda) d\lambda \quad \text{Eq. 20}$$

where DC_i , g_i and b_i are the digital count recorded by the detector, gain of the detector, and bias for the i^{th} spectral band, $\beta_i(\lambda)$ is the detector's spectral responsivity, and L_i is the effective radiance “seen” by the sensor.

Note that the measured radiance is affected by the spectral response of the sensor. Combining equations 18 and 20,

$$\begin{aligned} L_i &= \int_{\lambda} \sum_e^N L_e(\lambda) f_e \beta(\lambda) d\lambda \\ L_i &= \sum_{e=1}^N f_e \int_{\lambda} L_e(\lambda) B_i(\lambda) d\lambda \\ L_i &= \sum_{e=1}^N L_{e,i} f_e, \quad i = 1 \dots k \end{aligned} \quad \text{Eq. 21}$$

where $L_{e,i}$ is the effective radiance of endmember e measured in the i^{th} band of the sensor, and k is the number of bands.

The effects of the atmosphere can often be removed. Green *et al* (1993) present a method of calibrating AVIRIS data to eliminate atmospheric effects. When digital counts are corrected for atmospheric effects, spectral mixture analysis may be done in terms of the apparent reflectance of the endmembers

$$\begin{aligned} DC_i &= g_i \cdot R_i + b_i \\ R_i &= \sum_{e=1}^N R_{e,i} f_e \end{aligned} \quad \text{Eq. 22}$$

where $R_{e,i}$ is the effective reflectance of endmember e in the i^{th} spectral band.

Spectral mixture analysis produces equivalent results if calculations are performed in terms of radiance (equation 21) or reflectance (equation 22). The form of equation 22 will be used for the remaining discussion.

Spectral mixture analysis is performed using the following equation

$$\text{LRXS}_i = \sum_{e=1}^N R_{e,i} f_e + \varepsilon_i \quad i = 1 \dots k \quad \text{Eq. 23}$$

where LRXS_i is the digital count in the i^{th} spectral band, f_e is the unknown fraction of endmember e in the pixel, $R_{e,i}$ is the reflectance of reference endmember e in the i^{th} band (obtained from a library of endmembers), ε_i is the error in band i for the fit of N endmembers, and k is the number of bands in the low-resolution image. Spectral mixture analysis requires that the least squares fit to equation 23 be “good” (A good fit occurs when the RMS of the ε_i values is approximately the same magnitude as the sensor noise). The error is due to the residual variance, and is a measure of the spectral variation not predicted by the model.

The goal is to calculate the N unknown fractions. Equation 23 is the available equation and provides a constraint on the number of endmembers that can be unmixed or the number of bands required in the multispectral image. So the number of endmembers must be $k \geq N$. Using LANDSAT TM as an example, the maximum number of endmembers that can be unmixed is 6 ($k = 6$). This is a rather small number of possible endmembers and explains why hyperspectral sensors are much more suited to unmixing applications than multispectral sensors. The multiple bands of the hyperspectral sensor (e.g. 224 bands for AVIRIS) are ideal for use with unmixing equations.

Smith *et al* (1990) use a two-step process where the image is modeled as mixtures of image derived endmembers and then the image endmembers are modeled as mixtures of reference endmembers. Image endmembers are often a mixture of other materials and are selected such that a minimum number of reference spectra combine to describe them. For example, an image endmember may actually be composed of 40% vegetation and 60% soil because no pure pixels of vegetation or soil are present in the scene. In the second step of the process the fractions of this image endmember would be unmixed into fractions of reference spectra for soil and vegetation. Image endmembers are expressed as linear mixtures of reference endmembers in the same way that image data is expressed as mixtures of image endmembers. The spectra of the reference endmembers

are convolved with the bandpass of the sensor bands to ensure accurate comparison between the image endmembers and the reference endmembers.

Smith *et al* (1990) use a shade endmember to account for shading and shadows. The fraction image basically reflects lighting and topography variations in the image. To compensate for possible anticorrelation between vegetation/soil fractions and the shade fractions, all fractions except shade are re-scaled to sum to unity, pixel by pixel. For example, a vegetation endmember may be scaled using

$$Vf_s = \frac{f_{veg}}{(1 - f_{shade})} \quad \text{Eq. 24}$$

where Vf_s is the scaled vegetation fraction, f_{veg} is the original vegetation fraction, and f_{shade} is the shade fraction. This process removes only the shade fraction from the pixel. The scaling is correct assuming shade is equally present among all the endmembers. For display purposes, the complement of the shade image ($1-f_{shade}$) is combined with the Vf_s image to produce an image which matches observer intuition (e.g. high shade fractions appear dark).

If two or more endmembers are closely related (e.g. different types of soils), the same procedure for normalizing for shade can be used to emphasize the fractions between these closely related endmembers. For example, given two different soil endmembers, (Sa and Sb), a scaled fraction for soil endmember Sa can be generated by

$$Saf_s = \frac{f_{Sa}}{(f_{Sa} + f_{Sb})} \quad \text{Eq. 25}$$

where Saf_s is the scaled fraction for endmember Sa. Higher values of Saf_s indicate an abundance of soil type a, and low fractions indicate more of soil type b. This process can be used with equation 22 to produce fraction maps that are more closely calibrated to ground measurements.

2.2.3 Constraint Conditions

The previous discussion presented spectral mixture analysis as an unconstrained problem. However, the literature contains three different constraint conditions. The first is *unconstrained*, as previously presented, where fractions may assume whatever value is needed to produce an estimate with minimum error. The second condition is called *partially constrained*. Here, the sum of all the fractions within a pixel must be unity.

$$\sum_{e=1}^N f_e = 1 \quad \text{Eq. 26}$$

providing one equality constraint. Positive and negative fractions may be generated by both unconstrained and partially constrained unmixing. The *fully constrained* condition levies the additional requirement that all individual fractions lie between zero and one.

$$\sum_{e=1}^N f_e = 1, (0 \leq f_e \leq 1) \quad \text{Eq. 27}$$

providing $2 \cdot N$ inequality constraints.

Although the fully constrained situation seems to be the best method because it matches intuition, the negative fractions returned by the partially constrained case do have physical explanations. The following example may illustrate this point. Figure 9 illustrates mixtures of three materials in two spectral bands. The reflectance in each band is plotted along the axes. The vertices of the triangle are located at the reflectance of pure pixels of the three materials. Mixtures of the three materials with positive fractions are located along and within the perimeter of the triangle. For example, a 50/50 mixture of materials 1 and 2 materials lies midway between the vertices of these two materials as shown by the “X” in Figure 9.

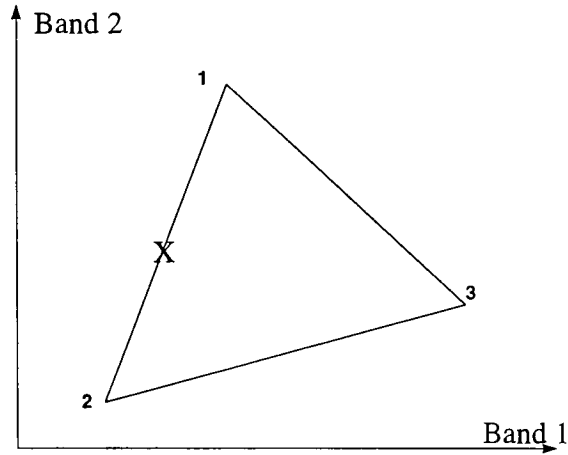


Figure 9: Three Material Mixtures in Two Spectral Bands

Figure 10 illustrates the results due to random variation. Although endmembers are plotted as specific points, they truly represent the *mean vectors*. If the real materials are gaussian distributed about the mean, then the contours plotted in Figure 10 represent equally likely departures from the mean values.

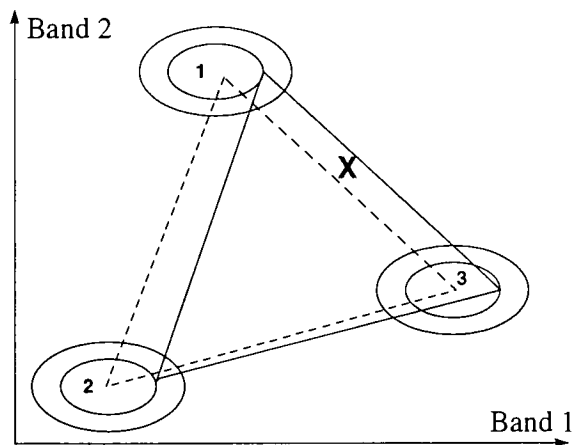


Figure 10: Mixture Requiring Negative Fractions

Suppose a particular pixel contains endmembers whose reflectance values are represented by the vertices of the solid triangle. The possible mixtures *for this pixel* lie within the solid triangle. Note that the point denoted by “X” lies within the solid triangle but outside the triangle formed by the reference endmembers. The mixture is a valid one, but requires negative fractions involving the reference endmembers.

2.2.3.1 Spectral Mixture Analysis Summary

Smith *et al* (1990) use Landsat TM data for spectral mixture analysis. The relatively small number of bands does not allow unique spectral identifications. Many materials, measured through the TM bandpass filters are indistinguishable from many mixtures of reference endmembers. Typically, there is no unique set of endmembers which combine to match the multispectral data. This is similar to the metamer found in color science. Spectral unmixing works best when applied to many bands of data as in the high spectral resolution of hyperspectral imagery. Spectral mixing is best applied when there is no interaction between scene elements (i.e. the linear mixing model applies). When non-linear mixing is present, other methods must be employed. The shade endmember can account for shading and topographic conditions. Since much of the variance in TM images is due to shading and shadows, the complement of the shade image can approximate the topography of a scene.

2.3 Image Fusion Via Stepwise Unmixing and Sharpening

Gross (1996) proposes an improved image fusion method based on stepwise regression. A low-resolution multispectral image is unmixed, producing low-resolution material maps. Conventional unmixing assumes that the number of endmembers exists throughout the entire image, attempting to find fractions for N

endmembers in every pixel. Gross implements a new method which adaptively estimates the endmembers within *each pixel*, and solves for the fractions for the n endmembers within the target pixel.

2.3.1 Stepwise Unmixing

The stepwise method requires that for each pixel, a library of L endmembers be searched for the n endmembers that are in that pixel. These endmembers are those that minimize the error. The output is the fractions for the n endmembers for the target pixel. In general terms, a predictive equation of the form

$$\hat{y} = Ax \quad \text{Eq. 28}$$

is used, where \hat{y} is the estimated spectral vector for the pixel, A is the matrix of reflectance values, and x is a vector containing the fractions. The main difficulty in stepwise regression is that n , the number of endmembers to be unmixed within a superpixel, is unknown. If n is chosen to be too large, over-fitting occurs and the solution tracks the noise in the data. Not only must the correct number of endmembers be used, but the most appropriate endmembers must be used as well. This could be done employing a search through L of all the possible combinations, but this method is computationally prohibitive. Such a strategy involves searching through $\frac{L!}{n!(1-n)!}$ combinations to obtain the optimum endmembers. As the size of the library increases, this number grows large quickly, requiring large amounts of computer resources. The stepwise method employed by Gross offers a less computationally prohibitive method.

Consider the basic ANOVA table illustrated in Table 1. Such an ANOVA table is typically formed to analyze the variance in a predictive model into its component parts: one due to the relationship with the predictor variable(s), and one due to error. Define the model as in equation 28, and let y be an m -vector, x an n -vector, and A an $m \times n$ matrix. The first column in Table 1 contains the variation source. The second column contains the degrees of freedom for that source. The third column shows how to calculate the Sum of Squares

(SS). The degrees of freedom and the Sum of Squares are referred to as “corrected” because the mean of y is subtracted from the measurements. The fourth and fifth column show uncorrected degrees of freedom and how to calculate the SS in matrix notation. The Mean Square (MS) , in the final column, is calculated by dividing the SS by the appropriate degrees of freedom.

Source	df (corrected)	Sum of Squares (SS)	df (uncorrected)	Sum of Squares (SS) (Matrix Form)	Mean Square (MS)
Regression	n-1	$\sum (y_i - \bar{y})^2$ (corrected)	n	SSR = $x'A'y$	MS(Regression) \Rightarrow MSR MSR = SSR/(n-1)
Error	m-n	$\sum (y_i - \hat{y}_i)^2$	m-n	SSE = $y'y - x'A'y$	MS (Error) \Rightarrow MSE MSE = SSE/(m-n)
Total	m-1	$\sum (y_i - \bar{y})^2$ (corrected)	m	$y'y$ (uncorrected)	

Table 1: Basic ANOVA Table

If the regression model is a good one, and the errors are gaussian with zero mean, then the errors should be chi-square distributed (χ^2). If the regression model is poor, then the errors will not be chi-square distributed. A hypothesis test can be used based on the relationship that the ratio of two chi-square variables divided by their degrees of freedom has an F-distribution as in

$$\frac{\chi_m^2 / m}{\chi_n^2 / n} \sim F_{m,n} \quad \text{Eq. 29}$$

where m and n denote the degrees of freedom for the two chi-square variables.

Now consider the SSR and SSE. If the errors are gaussian, then SSR and SSE are χ^2 distributed

$$\frac{SSR}{n-1} = MSR \sim \frac{\chi_{n-1}^2}{n-1} \quad \text{Eq. 30}$$

$$\frac{SSE}{m-n} = \text{MSE} \sim \frac{\chi^2_{m-n}}{m-n} \quad \text{Eq. 31}$$

and the ratio of

$$F = \frac{\text{MSR}}{\text{MSE}} \quad \text{Eq. 32}$$

will follow an $F_{n-1, m-n}$ distribution. The MSR/MSE ratio is formed and compared to a tabulated F-statistic with $n-1$ and $m-n$ degrees of freedom at the desired confidence level. If the ratio is greater than the value in the F-statistics table, then the regression model is a good one. If the ratio is less than the value in the table, then the regression model is rejected (this model would not explain enough of the variance to justify using it) and a better model should be used.

Stepwise regression is based on an ANOVA calculation of the “Extra Sum of Squares” (Draper & Smith, 1981). In this method an n -term model is compared with an $(n-1)$ -term model to determine the significance (benefit) of adding the additional term. Define the reduced-order term as

$$\hat{y} = Wz \quad ; \quad z = (W'W)^{-1}W'y \quad \text{Eq. 33}$$

where z is an $(n-1)$ -vector and W is an $m \times (n-1)$ matrix. The SS and MS are calculated as shown in Table 2.

Source	df (uncorrected)	SS	MS
<i>Regression</i>			
Reduced Model	$n-1$	$x'A'y - z'W'y$	
Extra Term	1	$z'W'y$	$\text{MS}_{\text{extra_term}}$
<i>Error</i>	$m-n$	$y'y - x'A'y$	MSE
Total	m	$y'y$ (uncorrected)	

Table 2: Extra Sum Of Squares ANOVA Table

As with the previous ANOVA table, the sum of squares are χ^2 distributed. The ratio of $MS_{\text{extra_term}}/MSE$ is compared to the value in a F-statistic table with the appropriate degrees of freedom, at the desired confidence level. If the ratio is greater than the tabulated value, then the regression model is valid, and the more complex model is required. If the ratio is smaller, then the simpler model is retained. In practice, a F-statistics table is not used, and a fixed value of F-to-enter and F-to-remove is used regardless of the degrees of freedom in a particular model being examined.

2.3.1.1 Stepwise Unmixing Summary

Stepwise selection ensures that the finally selected subset contains the proper number and most appropriate endmembers from the reference library. This method can map a greater number of endmembers than traditional methods, and can also prevent extraneous fractions from being over-fit to the image noise.

2.3.2 Constraints

After the appropriate endmembers are selected, unmixing may be performed unconstrained (as previously described), or with constraints. If constrained unmixing is desired, the final answer is obtained through a *restricted least squares*, involving linear equality and inequality constraints.

2.3.2.1 Equality Constraints

Once the number of endmembers to be examined is selected, then the remaining constraints must be applied to solve the predictive equation 28. The solution is the one that minimizes the error

$$\varepsilon = (y - \hat{y})^2 \tag{Eq. 34}$$

subject to equality constraints, which reduce the number of free variables in the solution space. This least squares problem can be solved using linear algebra.

A linear algebra solution to the constrained least square problem is presented by Lawson & Hanson (1974):

Given an $m_1 \times n$ matrix C of rank p , an m_1 -vector d , an $m_2 \times n$ matrix A , and an m_2 -vector y , minimize $\|y - Ax\|$ subject to $Cx = d$.

The solution exists if and only if the constraint condition ($Cx = d$) is consistent. If consistency is assumed, then $n > p = \text{rank}(C)$. The solution to the least square equality (LSE) problem is performed in three stages:

1. A lower-dimensional unconstrained least square problem is derived from the original constrained problem.
2. The derived problem is solved.
3. The solution is transformed to the original coordinate system to obtain the solution of the original constrained problem.

See Appendix A for details on solving the LSE problem.

2.3.2.2 Inequality Constraints

Lawson & Hanson (1974) also present a solution to the linear least square problem with linear inequality constraints:

Given an $m \times n$ matrix G , an m -vector h , an $m_2 \times n$ matrix A , and an m_2 -vector y , minimize $\|y - Ax\|$ subject to $Gx \geq h$.

While equality constraints reduce the number of free variables in the least square problem, inequality constraints establish boundaries within the solution space. An iterative solution is required to identify *active* constraints and restrict those affected variables. On each iteration, the active constraints are treated as equality constraints and a minimum is derived as previously described for equality constraints. See Appendix B for details on solving the LSI problem.

2.3.3 Sharpening

Gross (1996) proposes a method where fractions contained within low-resolution fraction maps are spatially located to the resolution of a higher resolution image, through a process called *sharpening*. See Figure 11 for an illustration of sharpening.

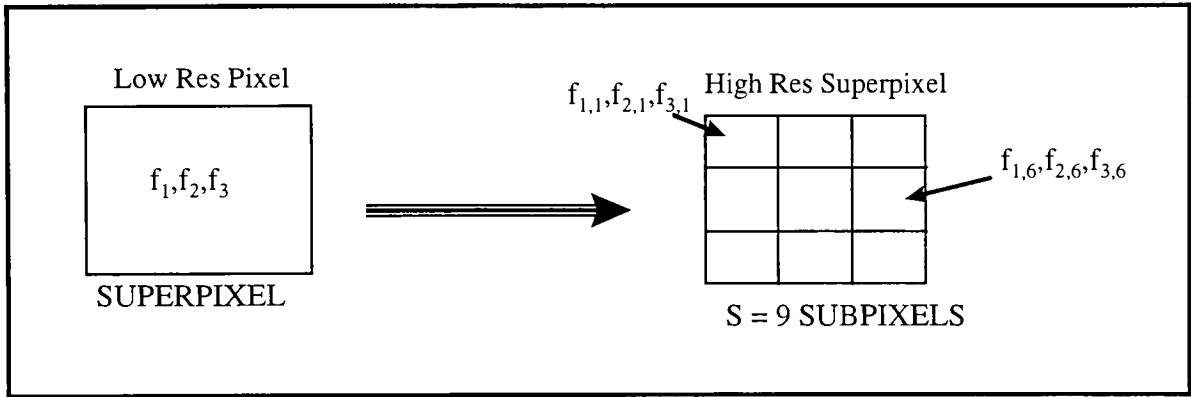


Figure 11: Illustration of Sharpening

The sharpening model has the same form as spectral mixture analysis.

$$HRP_j = \sum_{j=1}^n R_{pan,e} f_{e,j} + \varepsilon \quad j = 1 \dots s \quad \text{Eq. 35}$$

where HRP_j is the digital count in the i^{th} spectral band for the j^{th} subpixel of the high-resolution pan image, $R_{pan,e}$ is the reflectance of reference endmember e in the sharpening pan band(s), and $f_{e,j}$ is the high-resolution fraction of endmember e in subpixel j . This is a least squares problem, and $f_{e,j}$ is selected to minimize the error.

There is also a consistency requirement that the average of the high-resolution fractions for each endmember equal the original low-resolution fraction as

$$\frac{1}{s} \sum_{j=1}^s f_{e,j} = f_e \quad e = 1 \dots n \quad \text{Eq. 36}$$

where f_e is the original low-resolution fraction.

Sharpening may also be performed unconstrained (as previously discussed) or with constraints.

Partially constrained sharpening provides s equality constraints

$$\sum_{e=1}^n f_{e,j} = 1 \quad j = 1 \dots s \quad \text{Eq. 37}$$

However, only $(s-1)$ are independent. Fully constrained sharpening provides $2*n*s$ constraints

$$\sum_{e=1}^n f_{e,j} = 0, \quad (0 \leq f_{e,j} \leq 1) \quad j = 1 \dots s \quad \text{Eq. 38}$$

The equality constraints are not all independent. There are more unknowns than equations, so the sharpening model is solved as an under-determined least squares problem. The orthogonal decomposition method can be used to provide a solution. The fully constrained sharpening problem requires an iterative solution, and an optimization algorithm applies only the active constraints, as previously described for solving fully constrained unmixing problems.

Sharpening may be solved with Lagrange multipliers. Using Lagrange multipliers, the function

$F(x) = (y - \hat{y})^2$ is to be minimized subject to s equality constraints,

$$h_i(x) = c_i, \quad (i = 1 \dots s) \quad \text{Eq. 39}$$

An augmented function, called the Lagrangian, can be formed having the same minimum as $F(x)$

$$L(x) = F(x) + \lambda' (H(x) - C) \quad \text{Eq. 40}$$

where λ is a vector of Lagrange multipliers and the quantity $(H(x) - C)$ must be zero at the minimum. The minimum of L will also be the minimum of F . The requirements to minimize L are listed in Table 3.

$\frac{\partial}{\partial x} L = \frac{\partial}{\partial x} F(x) + \lambda' \frac{\partial}{\partial x} H(x) = 0$	Eq. 41
$\frac{\partial}{\partial \lambda} L = H(x) - C = 0$	Eq. 42

Table 3: Necessary Conditions to Minimize L

For the previously defined least squares problem, the function to be minimized is

$$F(x) = (y - \hat{y})^2 = (y - Rx)^2 \quad \text{Eq. 43}$$

subject to constraints,

$$H(x) = Hx - C \quad \text{Eq. 44}$$

where H is a s x n matrix, c is a p-element vector, and x is an n-element vector.

Recalling equation 40, then

$$\frac{\partial}{\partial x} L = -2R'(y - Rx) + H'\lambda = 0 \quad \text{Eq. 45}$$

and

$$\frac{\partial}{\partial \lambda} L = Hx - C = 0 \quad \text{Eq. 46}$$

In matrix form, this becomes

$$\begin{bmatrix} 2R'R & H' \\ H & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} = \begin{bmatrix} 2R'y \\ C \end{bmatrix} \quad \text{Eq. 47}$$

The first matrix in equation 47 can be inverted to solve for x and λ , producing

$$\begin{bmatrix} x \\ \lambda \end{bmatrix} = \begin{bmatrix} (R'R)^{-1} - WH(R'R)^{-1} & W \\ [H(R'R)^{-1}H']^{-1}H(R'R)^{-1} & -[H(R'R)^{-1}H']^{-1} \end{bmatrix} \begin{bmatrix} R'y \\ C \end{bmatrix} \quad \text{Eq. 48}$$

where $W = (R'R)^{-1}H^T[H(R'R)^{-1}H^T]^{-1}$, and the multiplier now accounts for the factor of 2. Then

$$\mathbf{x} = \mathbf{x}_u + \mathbf{W}(\mathbf{y} - \mathbf{H}\mathbf{x}_u) \quad \text{Eq. 49}$$

where $\mathbf{x}_u = (\mathbf{R}'\mathbf{R})^{-1} \mathbf{R}'\mathbf{y}$ is the estimator of high-resolution fractions.

2.3.3.1 Sharpening Summary

Sharpening takes the low-resolution fraction maps and enhances them, wherever possible, by spatially locating the materials to the same resolution as the sharpening band(s). It offers a way of improving the quality of the low-resolution fraction maps.

3. APPROACH

The available literature indicates that most image fusion algorithm techniques have only been evaluated in isolation, and that different routines have seldom been compared with each other. This research compared the results of the sharpen (fuse)/unmix procedure with those produced by the unmix/sharpen process. The unmix algorithms evaluated were the adaptive stepwise least squares method and the traditional least squares method.

3.1 Use of Synthetic Imagery (SIG)

Comparison of the two image fusion processes can be a difficult task. Therefore, it is useful to perform the procedures on a set of images whose radiometric, geometric, and spatial properties are known and can be controlled. Synthetic Image Generation (SIG) is ideal for such an application. The Digital Image Processing and Remote Sensing Synthetic Image Generation (DIRSIG) model used at RIT is such an image generation system. DIRSIG is a ray tracing algorithm which calculates radiometric signatures using a first principles approach. It can model such processes as upwelled and downwelled radiance, shadowing, and various interactions between scene elements and the environment (earth and sky) over the range from 0.28 to 20.0 μm . The user can construct scenes of varying complexity and also specify the sensor characteristics and responsivity. The output of the model is a scene which, in most cases, closely simulates the image (including spectral and spatial characteristics) that would be produced by a specific sensor under conditions provided by the user.

The previously mentioned image fusion processes were tested on imagery generated by DIRSIG. The complexity of the scenes was controlled by the user, increasing incrementally from simple, low spatial frequency images to complex high frequency ones used to test the final algorithms. Use of SIG allowed direct comparison of algorithm output to truth data. Varying the content of the scenes also provided an indication of the robustness of the algorithms and the optimum circumstances for applying them.

3.2 Test Method Overview

The algorithms were initially tested using SIG data. Low spatial frequency and high spatial frequency scenes were used to obtain optimum parameters for the algorithms. The resulting algorithms were validated by performing image fusion on real data. This final validation also ensured that there were no artifacts introduced from the parameter optimizations obtained through the SIG processing. One DIRSIG image was a forest scene, containing several camouflaged military vehicles. The other DIRSIG scene was an urban scene depicting downtown Rochester, NY. Two real scenes were obtained from the Western Rainbow data collection and are images of desert area containing military vehicles.

All original multispectral scenes were degraded by factors of 24X, 12X, and 6X. The derived panchromatic images were degraded by factors of 12X, 6X, 4X, and 3X. This resulted in images to be enhanced from 24X to 3X (a scale factor of 8X), from 24X to 4X (a scale factor of 6X), from 12X to 3X and 24X to 6X (a scale factor of 4X), and from 12X to 3X and 6X to 3X (a scale factor of 2X).

The process performed on each image is outlined in Figure 12. All fusion via the DIRS method used the global regression method, based on recommendations provided by Braun (1992). The fusion was performed at scale factors between the low-resolution and high-resolution imagery of 2X, 4X, 6X, and 8X. The resulting high-resolution image cubes were then unmixed via the stepwise method.

The low-resolution image cubes were unmixed using the stepwise and traditional methods. The low-resolution fraction maps were then sharpened using the high-resolution panchromatic data at sharpening scale factors of 2X, 4X, 6X, and 8X. The sharpened, unmixed material maps were compared to the unmixed, fused maps.

All unmixing was performed fully constrained, based on recommendations by Gross (1996). Fully constrained unmixing produces the least squared error (compared to unconstrained and partially constrained unmixing). All sharpening was performed using partial constraints. Gross states that fully constrained sharpening produces lower error, but the computational tradeoff outweighs the improved results.

All computer routines were written in the RSI's (Research Systems Incorporated) Interactive Data Language (IDL[®]) which is well suited for image processing routines.

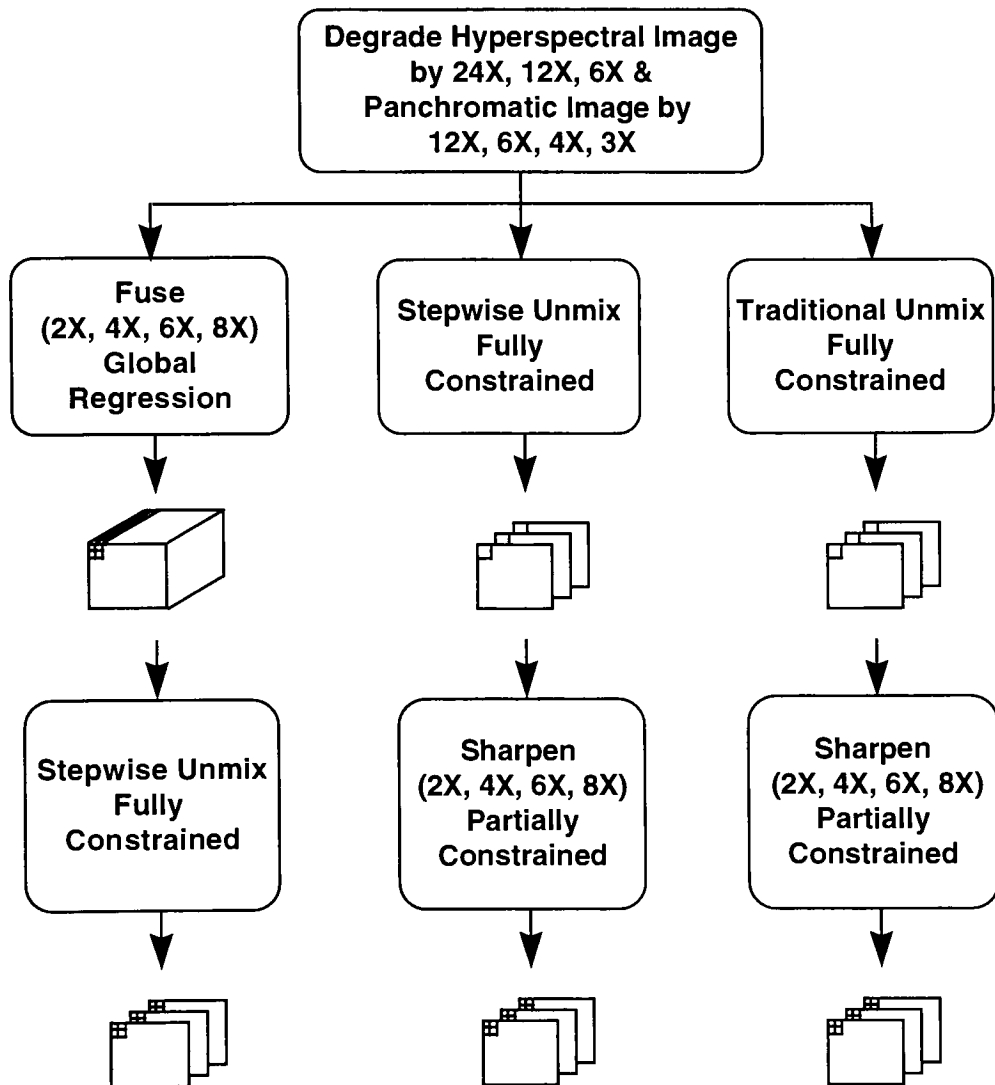


Figure 12: Test Plan Overview

3.3 Selection of Test Images

The first stage in performing validation was selection of final test scenes. Two DIRSIG scenes were used and two real scenes were used. The DIRSIG scenes were generated with bands which simulate the Environmental Research Institute of Michigan (ERIM) M-7 airborne line scanner(MUG, 1995). The fifteen bands of the M-7 sensor cover the visible (VIS), near infrared (NIR), and short wave infrared (SWIR) regions of the spectrum. The M-7 Sensor has “configurable” bands, and two different band configurations were simulated. The band passes for the various spectral bands are detailed in Table 4 and Table 5

Band Number	Low	High
1	0.44	0.5
2	0.46	0.53
3	0.495	0.57
4	0.46	0.62
5	0.58	0.675
6	0.615	0.72
7	0.66	0.765
8	0.705	0.93
9	0.76	1.045
10	0.9	1.385
11	1.1	1.39
12	1.3	1.79
13	1.4	1.89
14	1.9	2.39
15	1.9	2.49

Table 4: M-7 (Forest) Spectral Bands (μm)

The forest DIRSIG scene used was based on 15 band imagery contained in the Southern Rainbow data collection. Table 4 shows the bandpass data used for the forest scene. The spatial resolution of this image was approximately one meter, and contained 672 rows and 672 columns. A color version (using bands in the visible region) of this test scene is shown in Figure 13. The image contains deciduous trees, grass, several dirt roads, a small lake, and several small tanks and trucks, consisting of camouflage paints, canvas, and painted steel.

The other DIRSIG image used was an urban scene, depicting downtown Rochester, NY. This scene was originally used by White (White 1996) but the DIRSIG simulation was re-run to simulate the M-7 band passes. Table 5 shows the bandpass data used for the Rochester Scene. The spatial resolution of this image was approximately one meter, and contained 552 rows and 744 columns. A color version (using bands in the visible region) of this test scene is shown in Figure 14. The image contains deciduous trees, a river running along the bottom of a gorge, grass and loam, grass, and buildings and roads constructed with several types of brick, wood, shingles, concrete, and asphalt.



Figure 13: Forest Test Scene (Color Image)

Band Number	Low	High
1	0.45	0.47
2	0.48	0.5
3	0.51	0.55
4	0.55	0.6
5	0.6	0.64
6	0.63	0.68
7	0.68	0.75
8	0.73	0.81
9	0.81	0.92
10	1.02	1.11
11	1.21	1.3
12	1.53	1.64
13	1.54	1.75
14	2.08	2.2
15	2.08	2.37

Table 5: M-7 (Rochester) Spectral Bands (μm)



Figure 14: Rochester DIRSIG Scene (Color Image)

Two real scenes were used to evaluate the performance of the methods on actual data. Both were images from the Western Rainbow data collection. One image was obtained by the DEADALUS hyperspectral sensor. DAEDALUS is a 12-band sensor with band passes covering the VIS, NIR, SWIR, and thermal regions as shown in Table 6. The last two bands were not used for this work, so the scene used contained 10 bands. The image was acquired at an altitude of 250 ft and has a spatial resolution of approximately 1 meter. The final scene chosen contained 936 columns and 696 rows. Band 4 of this test scene is shown in Figure 15. The image contains desert pavement, silty soil, sparse vegetation, tanks, armored personnel carriers, and a mobile SCUD launcher.



Figure 15: Band 4 (570 - 650 nm) of DAEDALUS Image

Band Number	Low	High
1	0.405	0.455
2	0.435	0.535
3	0.5	0.625
4	0.57	0.65
5	0.595	0.72
6	0.645	0.79
7	0.7	0.955
8	0.785	1.07
9	1.495	1.835
10	2.011	2.56
11 (Not Used)	2.525	5.575
12 (Not Used)	7.6	14

Table 6: DAEDALUS Spectral Bands (μm)

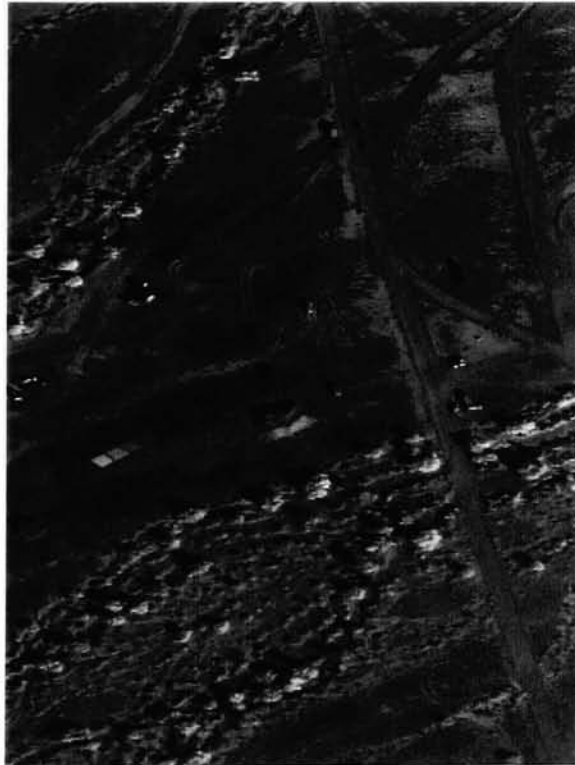


Figure 16: Band 15 (749 - 760 nm) of HYDICE Image

The second real scene used was obtained by the **Hyperspectral Digital Imagery Collection Experiment (HYDICE)** sensor. **HYDICE** is a hyperspectral sensor with 210 bands from 390 nm to 2500 nm. To reduce computational complexity, bands within atmospheric absorption regions were removed, and approximately every fourth band of those remaining was used for the final scene. The bandpasses of the resulting image are shown in Table 7. The test scene had a spatial resolution of approximately 1 meter, and contained 400 rows, 304 columns, and 44 bands. The test scene, shown in Figure 16, contains desert pavement, road, silty soil, sparse vegetation, test panels, and tanks, cars, and armored personnel carriers.

Band Number	Low	High	Band Number	Low	High
1	408.34	411.72	23	1191.50	1206.36
2	422.00	425.54	24	1250.76	1265.44
3	436.35	440.09	25	1309.23	1323.67
4	451.55	455.55	26	1436.86	1450.63
5	467.82	472.12	27	1491.58	1505.00
6	485.36	490.02	28	1531.75	1544.94
7	504.42	509.51	29	1584.16	1597.02
8	525.26	530.85	30	1635.28	1647.82
9	548.19	554.37	31	1685.12	1697.35
10	573.55	580.40	32	1733.73	1745.67
11	601.70	609.32	33	1792.85	1804.43
12	633.02	641.49	34	1970.70	1981.26
13	667.84	677.25	35	2043.64	2053.81
14	706.46	716.85	36	2084.11	2094.08
15	749.02	760.39	37	2123.79	2133.56
16	795.49	807.81	38	2162.65	2172.25
17	845.64	858.79	39	2200.80	2210.20
18	899.01	912.87	40	2238.22	2247.45
19	955.02	969.40	41	2284.04	2293.06
20	1012.94	1027.68	42	2319.95	2328.83
21	1072.10	1087.00	43	2363.99	2372.67
22	1131.80	1146.74	44	2398.56	2407.08

Table 7: HYDICE Spectral Bands (nm)

3.4 Creation of Panchromatic Data Sets

Rather than using DIRSIG to generate a new panchromatic band for the synthetic scenes, Several bands were used to generate high-resolution panchromatic images. Bands of the forest scene were combined using the following equation

$$PAN = \frac{\left\{ \begin{array}{l} 0.07 * \text{band 2} + 0.095 * \text{band 5} + \\ 0.105 * \text{band 7} + 0.975 * \text{band 9} + 0.49 * \text{band 12} \end{array} \right\}}{1.685} \quad \text{Eq. 50}$$

to produce a panchromatic band. The coefficients are obtained by using the bandpass of each band (the coefficient in the denominator is the sum of all the coefficients in the numerator).

A different band combination was used for the Rochester scene. The following equation

$$PAN = \frac{\left\{ \begin{array}{l} 0.02 * (\text{band 1} + \text{band 2}) + 0.04 * (\text{band 3} + \text{band 5}) + \\ 0.05 * \text{band 4} + 0.07 * \text{band 7} \end{array} \right\}}{0.24} \quad \text{Eq. 51}$$

to produce a visible panchromatic band from 450 nm to 750 nm.

A panchromatic sensor was not included in the Western Rainbow data set. Various bands of the hyperspectral images were used to generate panchromatic bands. For the DAEDALUS scene, bands 1,3, and 5 were used. The following equation

$$PAN \text{ Band} = \frac{0.05 * \text{band 1} + 0.125 * \text{band 3} + 0.125 * \text{band 5}}{0.30} \quad \text{Eq. 52}$$

was used to produce a visible panchromatic band from 405 to 720 nm.

A three-band sharpening image was created for the HYDICE image. The following equations were used

$$\begin{aligned}
\text{PAN1} &= \frac{\left\{ \begin{array}{l} 3.38 * \text{band 1} + 4.0 * \text{band 4} + 5.09 * \text{band 7} + 6.18 * \text{band 9} + \\ 7.62 * \text{band 11} + 8.47 * \text{band 12} + 10.39 * \text{band 14} + 11.37 * \text{band 15} \end{array} \right\}}{55.50} \\
\text{PAN2} &= \frac{\left\{ \begin{array}{l} 13.8 * \text{band 18} + 14.74 * \text{band 20} + \\ 14.94 * \text{band 22} + 14.68 * \text{band 24} + 14.44 * \text{band 25} \end{array} \right\}}{72.66} \\
\text{PAN3} &= \frac{\left\{ \begin{array}{l} 10.17 * \text{band 35} + 9.77 * \text{band 37} + \\ 9.4 * \text{band 39} + 9.02 * \text{band 41} + 8.88 * \text{band 42} \end{array} \right\}}{47.24}
\end{aligned} \tag{Eq. 53}$$

to produce a VIS panchromatic image from 408 nm to 760 nm, a NIR panchromatic image from 899 nm to 1324 nm, and a SWIR panchromatic image from 2043 nm to 2320 nm.

Typically, image fusion requires that the panchromatic image be registered to the multispectral image. This process involves selection of ground control points in both images, the application of some polynomial fit, and re-sampling via a routine such as nearest neighbor, or bilinear interpolation. Since the panchromatic bands used in this research were derived from the original multispectral images, the LRXS and HRP images were perfectly registered, and geometric registration was not required

3.5 Generating Fraction Maps

Since the true elements in the DIRSIG scenes were known, true fraction maps were created. These perfectly unmixed material maps were compared to the results of the unmix and sharpen algorithms. Samples showing material maps from the forest scene for grass, dirt, and deciduous trees (degraded to 3X) are shown in Figure 17. Sample fraction maps of the Rochester scene for grass, water, roof gravel, loam, trees, new and old asphalt, concrete, glass, and shingles are shown in Figure 18. Recall that the digital count in the material map is proportional to the fraction of that material for each pixel location.



Figure 17: Perfectly Unmixed Material Maps for Grass, Dirt, & Deciduous
Trees (Forest Scene)

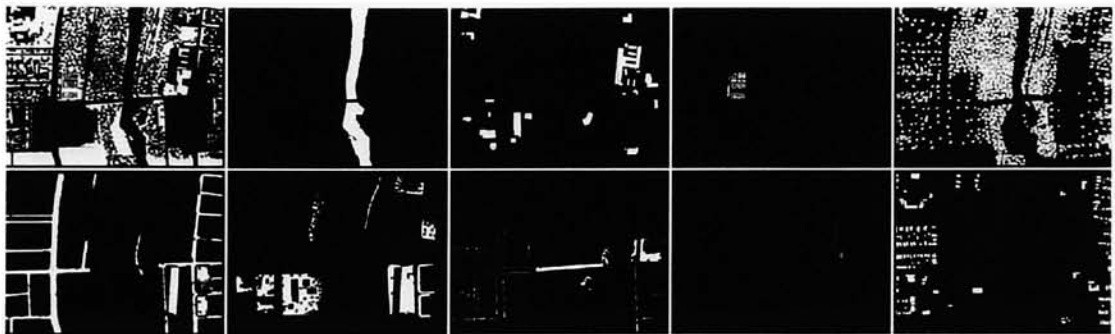


Figure 18: Perfectly Unmixed Material Maps for Grass, Water, Roof Gravel, Loam, Trees,
New Asphalt, Old Asphalt, Concrete, Glass, and Shingles (Rochester Scene)

3.6 Endmember Selection

Two options are available for selecting spectral libraries. Reference spectra from a library of endmembers, or scene-derived endmembers may be used. The reference spectra are often recorded in laboratory conditions, or in the field. In either situation, the measured radiance does not travel through several thousand feet of atmosphere, so the spectra are much sharper for these reference spectra than those measured by the

sensor. In order for the reference spectra to be employed, the effects of the atmosphere must be removed from the image as described by Green (1993). To avoid this step, scene-derived endmembers were used.

To obtain these endmembers, “pure” pixels were located within the images. A Maximum Noise Fraction (MNF) transform was first performed to reduce the dimensionality of the images. The resulting MNF images served as the input for the Pixel Purity Index (PPI). The result of the Pixel Purity Index was a map of pure pixels which were likely candidates for mixture endmembers. The details of the MNF transform and Pixel Purity Index are described below.

3.6.1 Maximum Noise Fraction (MNF) Transform

The Maximum Noise Fraction Transform, described by Green (Green et al 1988) can determine the inherent dimensionality of data and segregate noise in the data. The MNF transform is designed to be an improved alternative to the Principal Component Transform.

The Principal Component Transform uses a linear transformation to translate and rotate data into a new coordinate system that maximizes the variance. This transformation is useful for enhancing information content, compressing useful image information into the low-order Principal Components. This compression is evidenced by a steady decrease in signal-to-noise ratio as the Principal Component number increases. However, this trend does not appear in all data sets, and sometimes a higher Principal Component can contain more useful data than some preceding (lower) ones.

The MNF Transform is designed to prevent such an occurrence. It is basically two cascaded Principal Component Transforms. The first transformation is based on an estimate of the noise covariance matrix and decorrelates the noise in the data. This step exploits the fact that the signal in a pixel is strongly correlated with the signal of neighboring pixels, while the noise shows weak spatial correlation. The result of this step is a noise-whitened data set, in which the noise has unit variance and no band to band correlations. The second step is a Principal Components Transformation of the noise-whitened data. The full data space can then be divided

into two parts: coherent, spatially meaningful images (with large eigenvalues), and noise-dominated images (with near unity eigenvalues).

The coherent images serve as a data set essentially free of noise which can be used for further processing with reduced computational complexity. The MNF Transform was implemented by the ENVI[®] image processing software by Research Systems Incorporated (RSI). The coherent bands of the MNF image served as input for the Pixel Purity Index.

3.6.2 Pixel Purity Index

The Pixel Purity Index (PPI) is a method to find spectrally pure pixels in multispectral and hyperspectral images. This is done with the assumption that spectrally pure pixels are likely to correspond to mixing endmembers. The Pixel Purity Index is computed by randomly generating a unit vector and projecting the n-dimensional scatterplots of image data onto the unit vector. Next, the pixels that project at the extremes are identified, and a record is kept of the total number of times each pixel is marked as extreme. See Figure 19 for an example in two dimensions. The resulting image is a brightness map which records how often a pixel was defined to be extreme. Thresholding can be performed on the PPI image to identify pixels or clusters that are candidate endmembers.

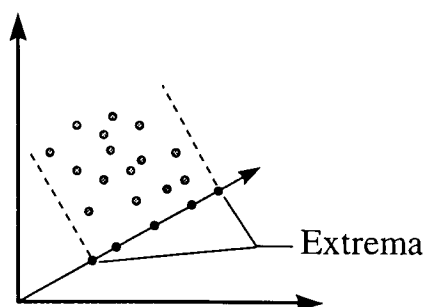


Figure 19: Locating Extrema in the Pixel Purity Index

The Pixel Purity Index can be implemented on Maximum Noise Fraction (MNF) transformed images to reduce the amount of data and to minimize the variation in extremes due to noise. This is suitable since the PPI image is only a map of pixel locations of candidate endmembers.

The PPI data can be plotted, and clusters of pixels indicate candidate endmembers. This normally requires interactive selection and the use of multiple projections of the n-dimensional data onto 2-D plots to define endmembers. See Figure 20 for a 2-D example.

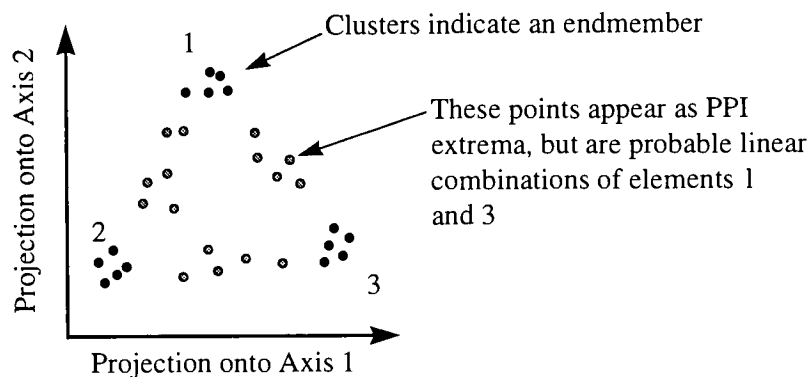


Figure 20: Obtaining Endmembers from PPI Clusters

Once the endmembers are defined in these projections, the corresponding pixels can be used to define the endmember vectors from the means of the selected pixels in each cluster in either reflectance or radiance space.

The process of obtaining the scene-derived endmembers was rather labor intensive. The “pure” pixels indicated by the PPI image did not always produce acceptable results. The spectrum of a material was obtained by using the PPI image as a mask within a region of interest and obtaining the average of the highlighted pixels in the multispectral images. A trial unmixing was then performed. If the resulting fraction maps were visually acceptable, then the spectrum was retained in the library. This was easily done with the DIRSIG scenes because

class maps could be used to provide quantitative data to gauge performance. The real scenes used for this research did not contain much clutter and the success of the unmixing for the candidate spectrum could be determined simply by looking at the higher resolution original image. This would have been a more difficult process if the military vehicles had been camouflaged. Certain band combinations were especially useful in highlighting materials (e.g. vegetation) in the original image for comparison with the class maps. If the PPI pixels produced bad results, nearby pixels were used instead to obtain the library.

The spectra used to unmix the images are presented in both tabular and graphical form. Please see Tables 8 through 11 and figures 21 through 24.

Band	Shadow	Camo Paint	Water	Grass	Rubber	Dirt	Tree Bark	Glass	Trees	Steel Side	Steel Bumper	Canvas
1	0.2840	0.2996	0.3098	0.3093	0.2675	0.2573	0.2725	0.4157	0.2843	0.3358	0.2492	0.2686
2	0.2824	0.3030	0.3100	0.3135	0.2663	0.2586	0.2750	0.4275	0.2892	0.3308	0.2462	0.2682
3	0.2933	0.3184	0.3158	0.3290	0.2690	0.2753	0.2949	0.4471	0.3220	0.3298	0.2517	0.2728
4	0.2872	0.3236	0.3128	0.3390	0.2666	0.2763	0.3051	0.4588	0.3157	0.3233	0.2486	0.2711
5	0.2807	0.3259	0.3107	0.3515	0.2677	0.2752	0.3105	0.4706	0.2973	0.3229	0.2513	0.2736
6	0.2785	0.3270	0.3052	0.3546	0.2697	0.2720	0.3147	0.4784	0.2909	0.3224	0.2538	0.2760
7	0.3134	0.3372	0.2853	0.3887	0.2631	0.3500	0.3924	0.4745	0.3717	0.3218	0.2568	0.2688
8	0.4118	0.3863	0.3098	0.4911	0.2980	0.5024	0.5093	0.5216	0.5087	0.3683	0.3028	0.3052
9	0.3877	0.3518	0.2706	0.4723	0.2689	0.5020	0.5061	0.4863	0.4817	0.3419	0.2783	0.2775
10	0.3989	0.3459	0.2625	0.5183	0.2732	0.5557	0.5434	0.5020	0.4856	0.3556	0.2886	0.2850
11	0.3812	0.3310	0.2510	0.5164	0.2647	0.5321	0.5191	0.5059	0.4610	0.3620	0.2793	0.2787
12	0.3373	0.3295	0.2628	0.4795	0.2784	0.3826	0.4059	0.5255	0.3945	0.3904	0.2750	0.2912
13	0.3273	0.3190	0.2549	0.4652	0.2700	0.3826	0.3944	0.5059	0.3816	0.3816	0.2681	0.2823
14	0.2572	0.2855	0.2382	0.3956	0.2511	0.2691	0.2887	0.4824	0.2838	0.3788	0.2397	0.2648
15	0.2766	0.3020	0.2601	0.3957	0.2726	0.2899	0.3015	0.4745	0.2964	0.3892	0.2632	0.2835

Table 8: Forest Spectral Library

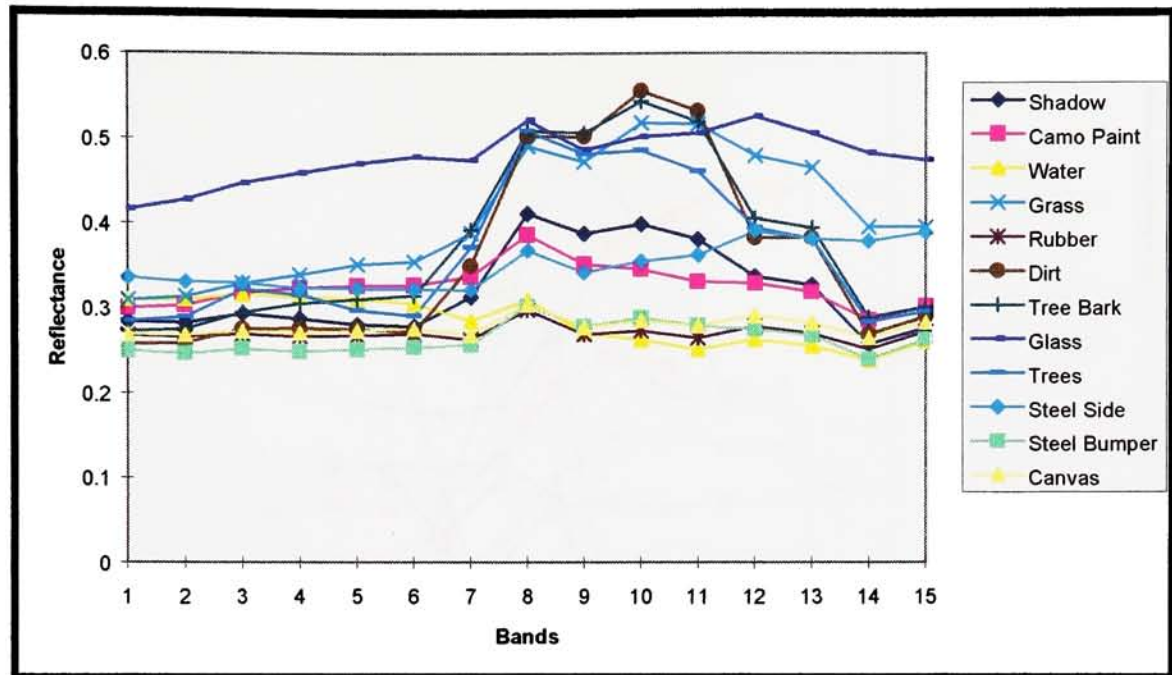


Figure 21: Forest Spectral Curves

Band	Grass	Water	Roof Gravel	Loam	Trees	New Asph	Old Asph	Concr	Glass	Shingle	Brick	Wood (Blue)	Wood (Red)	Wood (Gm)	Wood (Gray)
1	0.0593	0.0015	0.2273	0.0887	0.0912	0.1013	0.3399	0.4619	0.0540	0.7305	0.0543	0.2962	0.1169	0.0908	0.5389
2	0.0587	0.0088	0.2116	0.0903	0.0915	0.0985	0.3178	0.4313	0.0544	0.6135	0.0535	0.2846	0.1477	0.1433	0.5362
3	0.0692	0.0081	0.2002	0.1043	0.1704	0.0981	0.2984	0.4070	0.0557	0.5362	0.0558	0.2812	0.2485	0.2756	0.5427
4	0.0831	0.0021	0.1968	0.1163	0.1723	0.0952	0.2691	0.3643	0.0530	0.4754	0.0653	0.2561	0.2559	0.1984	0.5152
5	0.0624	0.0000	0.2069	0.1325	0.1149	0.0956	0.2620	0.3467	0.0504	0.4568	0.0880	0.2428	0.2616	0.1039	0.5052
6	0.0500	0.0000	0.2024	0.1434	0.0780	0.0945	0.2567	0.3353	0.0490	0.4441	0.0917	0.2359	0.2597	0.0856	0.4940
7	0.0870	0.0000	0.2176	0.1703	0.2033	0.0996	0.1525	0.3276	0.0538	0.2284	0.1076	0.2303	0.2585	0.0967	0.4787
8	0.3027	0.0000	0.2689	0.2418	0.4257	0.1322	0.0544	0.3381	0.0794	0.0029	0.1578	0.2348	0.2641	0.1132	0.4854
9	0.6153	0.0000	0.3606	0.4273	0.6589	0.2074	0.0839	0.4458	0.1247	0.0052	0.2441	0.1981	0.2181	0.1081	0.4310
10	0.6230	0.0000	0.4056	0.5501	0.6254	0.2123	0.0874	0.4020	0.1252	0.0036	0.2961	0.0179	0.0130	0.0175	0.0337
11	0.5448	0.0000	0.3885	0.5841	0.5212	0.2018	0.0863	0.3553	0.1591	0.0025	0.3036	0.0159	0.0114	0.0158	0.0308
12	0.3463	0.0000	0.3577	0.6213	0.3506	0.2111	0.0945	0.3324	0.1435	0.0023	0.2805	0.0139	0.0094	0.0133	0.0286
13	0.3674	0.0000	0.3587	0.6213	0.3570	0.2125	0.0947	0.3311	0.1450	0.0023	0.2819	0.0140	0.0095	0.0134	0.0286
14	0.1774	0.0000	0.0037	0.6252	0.1831	0.2271	0.1038	0.3085	0.1310	0.0048	0.2608	0.0140	0.0099	0.0128	0.0323
15	0.1783	0.0000	0.0852	0.6157	0.1612	0.2244	0.0994	0.3034	0.1287	0.0022	0.2601	0.0127	0.0083	0.0122	0.0286

Table 9: Rochester Spectral Library

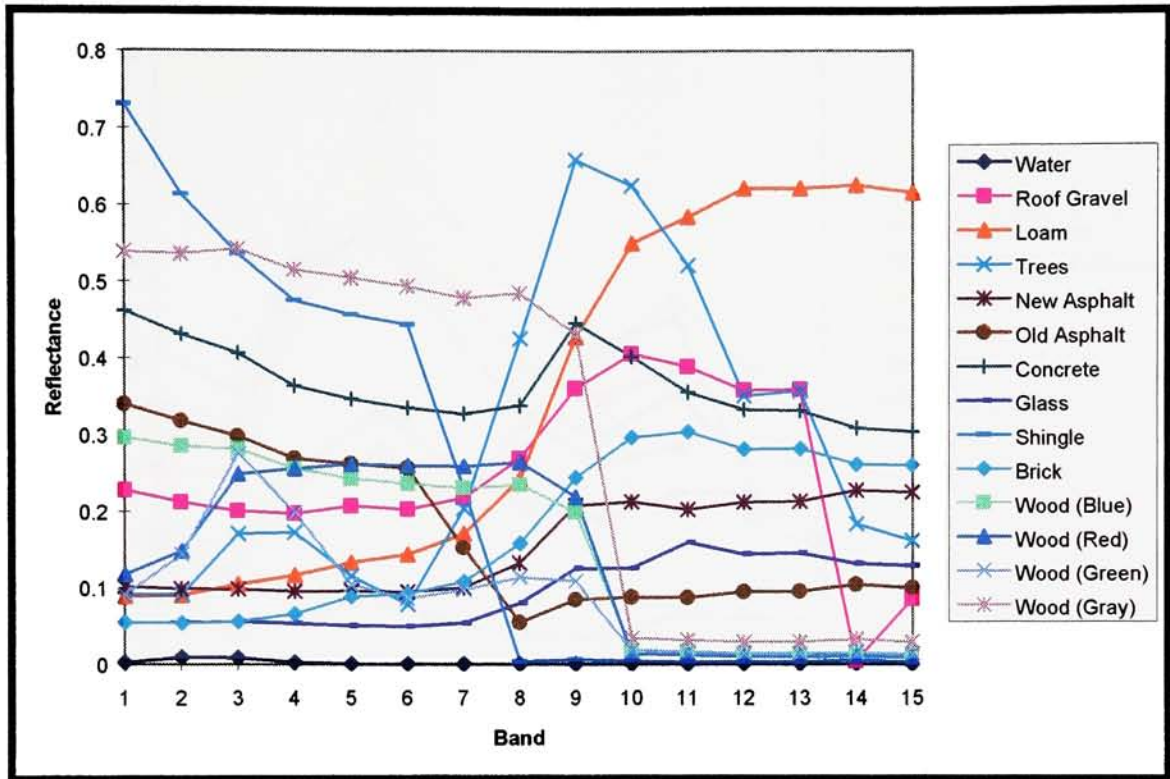


Figure 22: Rochester Spectral Curves

Band	Disturbed Desert							
	Desert Pavement	Pavement	Road	Silty Sand	Vegetation	Steel	Shadow	Canvas
1	0.1364	0.1967	0.2289	0.2620	0.1249	0.2226	0.0765	0.3110
2	0.1227	0.1908	0.2302	0.2426	0.0969	0.2244	0.0461	0.2736
3	0.2032	0.3596	0.4094	0.4409	0.1899	0.3917	0.0544	0.4030
4	0.2561	0.4603	0.4895	0.5457	0.2162	0.4312	0.0757	0.4274
5	0.2003	0.3711	0.3840	0.4366	0.1580	0.3268	0.0461	0.3250
6	0.1612	0.2963	0.2980	0.3452	0.2476	0.2436	0.0360	0.2515
7	0.1475	0.2636	0.2585	0.3090	0.3176	0.2023	0.0338	0.2306
8	0.1743	0.2942	0.2789	0.3641	0.3429	0.2183	0.0560	0.2706
9	0.1753	0.2505	0.2501	0.2887	0.1731	0.1496	0.0523	0.1992
10	0.1249	0.2085	0.1845	0.3354	0.0902	0.1172	0.0505	0.2056

Table 10: DAEDALUS Spectral Library (μm)

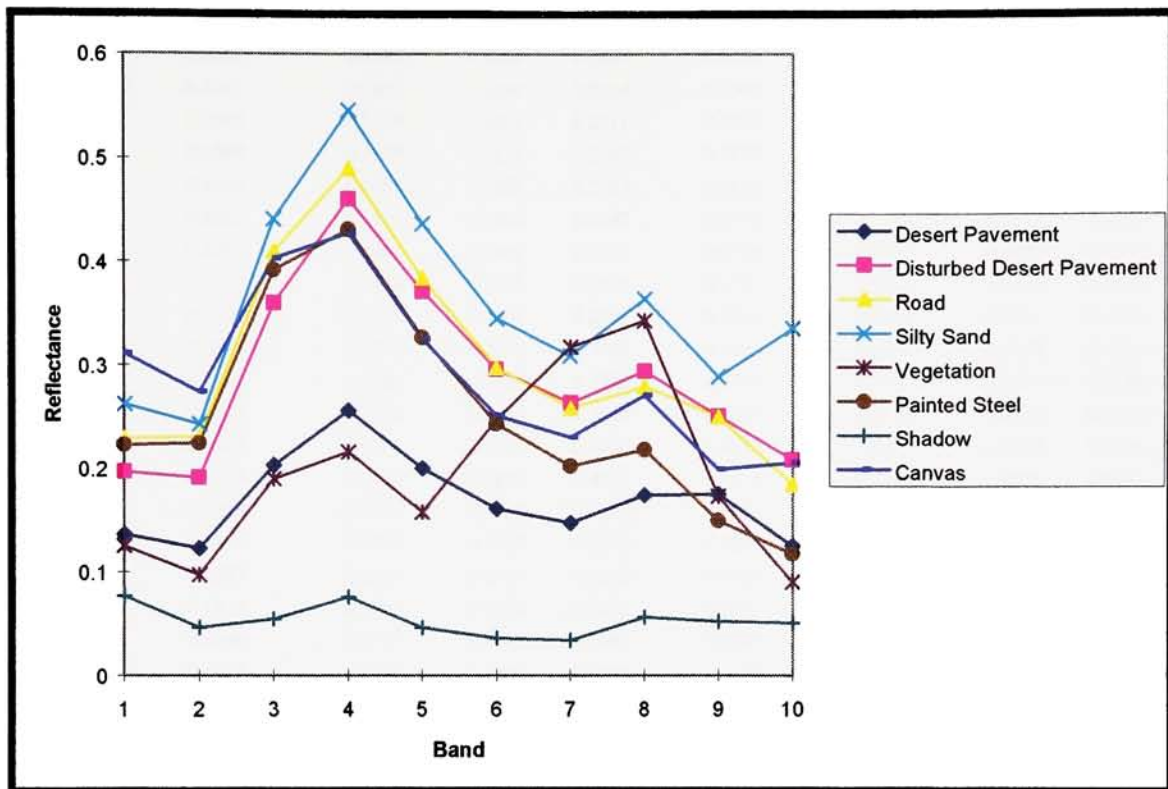


Figure 23: DADEALUS Spectral Curves

Band	Desert Pavement	Disturbed Pavement	Road	Silty Sand	Vegetation	Painted Steel	Shadow	Canvas
1	0.1025	0.1235	0.1683	0.2067	0.0456	0.6993	0.0573	0.5137
2	0.1021	0.1262	0.1694	0.2024	0.0500	0.7242	0.0521	0.4804
3	0.0989	0.1249	0.1671	0.2011	0.0529	0.6915	0.0456	0.4431
4	0.1059	0.1377	0.1821	0.2143	0.0623	0.6980	0.0483	0.4412
5	0.1043	0.1387	0.1803	0.2163	0.0623	0.6850	0.0445	0.4373
6	0.1083	0.1457	0.1844	0.2242	0.0652	0.6778	0.0452	0.4392
7	0.1072	0.1461	0.1839	0.2277	0.0740	0.6562	0.0420	0.4353
8	0.1135	0.1574	0.2003	0.2529	0.1157	0.6915	0.0420	0.4627
9	0.1179	0.1694	0.2109	0.2767	0.1304	0.6771	0.0431	0.4549
10	0.1349	0.1970	0.2346	0.3208	0.1064	0.6758	0.0480	0.4804
11	0.1532	0.2284	0.2640	0.3792	0.0951	0.7150	0.0474	0.5078
12	0.1611	0.2423	0.2724	0.3968	0.0843	0.7078	0.0522	0.5118
13	0.1626	0.2474	0.2702	0.4018	0.0623	0.6922	0.0500	0.5078
14	0.1736	0.2711	0.2887	0.4313	0.2716	0.7320	0.0552	0.5451
15	0.1776	0.2847	0.2966	0.4427	0.7574	0.7183	0.0649	0.6020
16	0.1850	0.2903	0.3027	0.4654	0.8221	0.7183	0.0668	0.6255
17	0.1847	0.2924	0.2984	0.4624	0.8343	0.7013	0.0673	0.6294
18	0.1833	0.2867	0.2884	0.4453	0.8015	0.6902	0.0659	0.6059
19	0.1840	0.2764	0.2783	0.4193	0.6549	0.6405	0.0610	0.5529
20	0.1775	0.2796	0.2806	0.4360	0.6779	0.6327	0.0621	0.5784
21	0.2040	0.3201	0.3180	0.4936	0.8245	0.6242	0.0731	0.6569
22	0.2027	0.3020	0.3004	0.4288	0.5951	0.5922	0.0676	0.5569
23	0.1819	0.2770	0.2776	0.4177	0.4525	0.5608	0.0588	0.5451
24	0.1818	0.2796	0.2813	0.4155	0.4828	0.5477	0.0588	0.5471
25	0.1851	0.2846	0.2872	0.4064	0.4181	0.5294	0.0604	0.5392
26	0.2437	0.3228	0.3283	0.4262	0.1216	0.4882	0.0811	0.5804
27	0.2009	0.2860	0.3071	0.4248	0.1127	0.4686	0.0450	0.5529
28	0.2009	0.2993	0.3246	0.4345	0.1412	0.4732	0.0463	0.5686
29	0.2016	0.2999	0.3251	0.4353	0.1809	0.4601	0.0485	0.5686
30	0.2078	0.3067	0.3312	0.4416	0.2108	0.4510	0.0510	0.5451
31	0.2210	0.3213	0.3478	0.4624	0.2167	0.4320	0.0535	0.5549
32	0.2211	0.3177	0.3423	0.4419	0.1941	0.4137	0.0521	0.5471
33	0.2627	0.3863	0.4024	0.4791	0.1966	0.4333	0.0930	0.6627
34	0.2721	0.3524	0.3595	0.4235	0.0598	0.3830	0.0728	0.6216
35	0.2752	0.3683	0.3928	0.4598	0.0716	0.3752	0.0722	0.6373
36	0.2724	0.3622	0.3879	0.4537	0.0789	0.3575	0.0651	0.5980
37	0.2772	0.3693	0.3930	0.4646	0.0951	0.3510	0.0659	0.5431
38	0.2791	0.3714	0.3905	0.4605	0.1064	0.3471	0.0676	0.5922
39	0.2755	0.3659	0.3599	0.4472	0.1206	0.3464	0.0767	0.6314
40	0.3098	0.4067	0.4093	0.4942	0.1328	0.3425	0.0921	0.5529
41	0.3637	0.4679	0.4836	0.5702	0.1358	0.3340	0.1111	0.6569
42	0.3402	0.4307	0.4328	0.5241	0.1265	0.3248	0.1103	0.5863
43	0.3505	0.4368	0.4250	0.5263	0.1436	0.3412	0.1376	0.6235
44	0.3589	0.4377	0.4250	0.5200	0.1495	0.3438	0.1526	0.6373

Table 11: HYDICE Spectral Bands (nm)

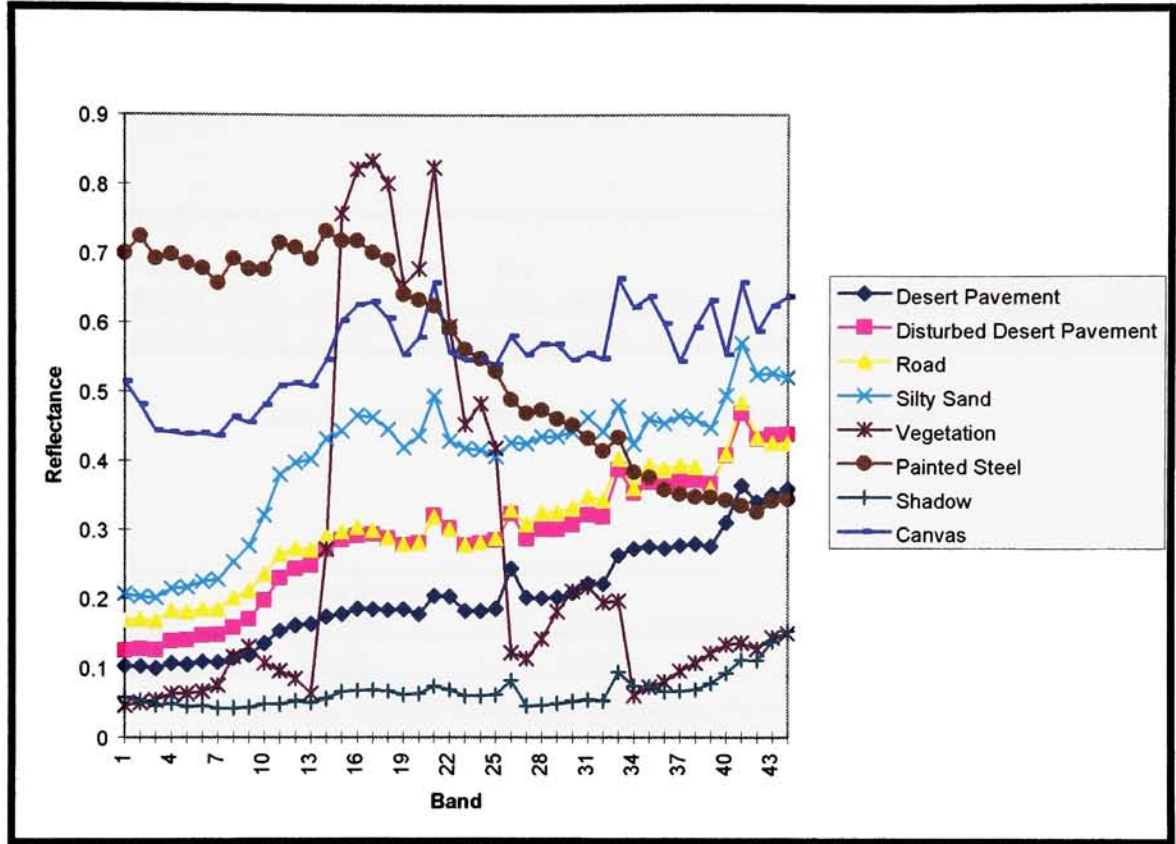


Figure 24: HYDICE Spectral Curves

3.7 Obtaining Sharpening Library

Sharpening required a reflectance library of the materials in the high-resolution panchromatic images. The sharpening library was obtained with the same equations used to generate the pan bands. For example, the pan library for the forest scene was generated using the reflectance value of each material (from the multispectral library) in bands 2, 5, 7, 9, and 12 as in equation 50. The sharpening libraries are shown in Tables 12 through 15.

Shadow	Camo Pant	Water	Grass	Rubber	Dirt	Tree Bark	Glass	Trees	Steel Side	Steel Bumper	Canvas
0.3491	0.3151	0.2708	0.4467	0.2717	0.3861	0.4534	0.5005	0.4313	0.3358	0.2536	0.2997

Table 12: Spectral Library for Sharpening Band (Forest)

Grass	Water	Roof Gravel	Loam	Trees	New Asph	Old Asph	Concr	Glass	Shingle	Brick	Wood (Blue)	Wood (Red)	Wood (Grn)	Wood (Gray)
0.0749	0.0016	0.2012	0.1292	0.1567	0.0978	0.1837	0.3461	0.0672	0.4323	0.0853	0.2573	0.2320	0.0807	0.5273

Table 13: Spectral Library for Sharpening Band (Rochester)

Desert Pavement	Disturbed Desert Pavement	Road	Silty Sand	Vegetation	Painted Steel	Shadow	Canvas
0.1908	0.3373	0.3687	0.4093	0.1658	0.3365	0.0546	0.3552

Table 14: Spectral Library for Sharpening Band (DAEDALUS)

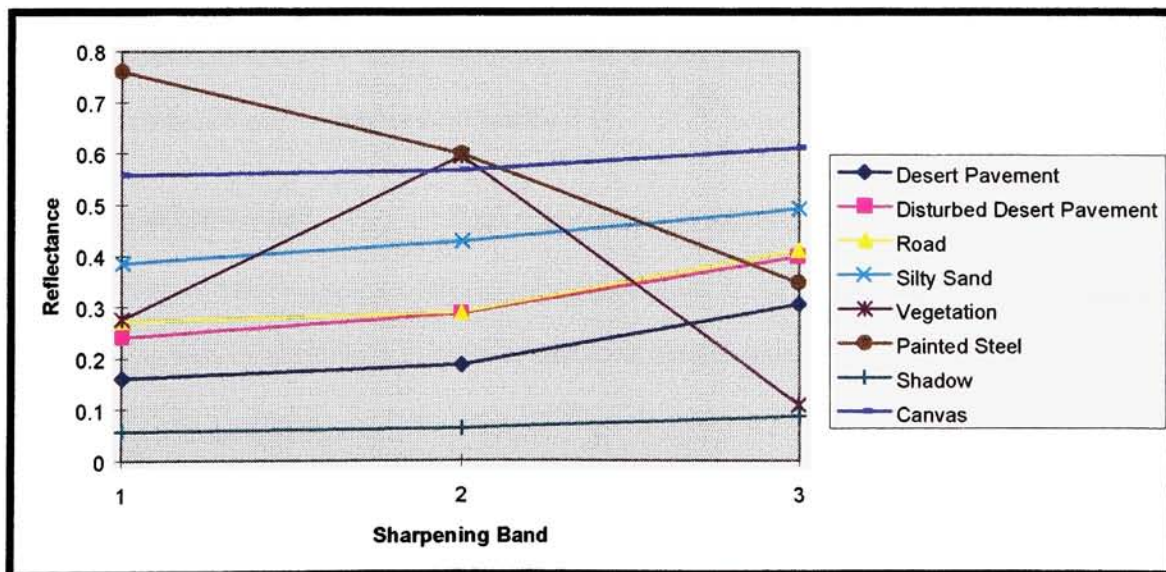


Figure 25: Spectral Curves for Sharpening Bands (HYDICE)

Band	Disturbed Desert					Painted		
	Desert Pavement	Pavement	Road	Silty Sand	Vegetation	Steel	Shadow	Canvas
1	0.1600	0.2401	0.2710	0.3846	0.2754	0.7600	0.0568	0.5570
2	0.1866	0.2872	0.2883	0.4273	0.5949	0.5991	0.0631	0.5665
3	0.3048	0.3988	0.4111	0.4914	0.1088	0.3471	0.0864	0.6108

Table 15: Spectral Library for Sharpening Bands (HYDICE)

3.8 Error Metrics

Error metrics were used to determine the efficiency of the algorithms. The fusion algorithm was evaluated by two metrics (effective RMS and effective edge RMS). These metrics allowed “validation” of the fusion routines by comparing average RMS values with those obtained by Braun (1992). The fraction maps were evaluated using the squared error metric proposed by Gross (1996). The squared error measurements were used to rank the output of the three image enhancement methods.

The images were also evaluated visually. Although a visual evaluation does not yield a numerical output, subjective observations were also helpful in evaluating the methods. When dealing with the real images, the visual evaluation was the only method available.

3.8.1 Squared Error

The use of SIG images provided an excellent opportunity to use a single error metric to compare the results. A squared error (SE) was calculated for each set of fraction maps.

$$SE = \frac{1}{N} \sum_{pixels} \sum_{materials} (f_{truth} - f_{test})^2 \quad \text{Eq. 54}$$

where N is the number of pixels in the image. The summation over the pixels included the entire image and was performed for the entire library of materials. Examination of the relative magnitude of this error provided a

measure of the match of the test fractions to the truth fractions. Errors of both commission and omission were measured by this metric.

3.8.2 Effective RMS

The efficiency of the fusion algorithm was judged by determining the root mean squared (RMS) difference between the fused image and higher resolution multispectral imagery. For example, the 16 m GIFOV multispectral images fused to 8 m GIFOV were compared to 8 m multispectral imagery.

The RMS error in the i^{th} band between the fused image and the truth image (ϵ_i) is given by

$$\epsilon_i = \sqrt{\frac{\sum_{n=1}^N (\text{HRXS}_i(n) - \text{Truth}_i(n))^2}{N}}, \quad i=1\dots k \quad \text{Eq. 55}$$

where k is the number of bands in the image. The final effective RMS for the image is the average RMS of all the bands

$$\text{RMS}_{\text{effective}} = \frac{1}{k} \sum_{i=1}^k \epsilon_i \quad \text{Eq. 56}$$

3.8.3 Effective Edge RMS

An effective RMS calculation was also performed in the vicinity of edges in the image. The edge finding routine was original code written by Dave Schlingmeier (1997). Edges are found using convolution of the image with Sobel, Roberts, Prewitt, Frei-Chen, and Laplacian operators. Several operators are used to prevent occurrences of isolated pixels, ensuring the resulting edge map contains mostly closed contours. The histogram of the summed-edge image is then used to determine a threshold. Values above the threshold are set to 255 and all others are set to 0. Finally, these edge pixels are grown into a 3x3 cube. The result is a binary edge mask where non-zero pixels occur within a two-pixel vicinity of edges. The edge mask is employed on the

HRXS output and “truth” images. The effective edge RMS was calculated as in equations 55 and 56 using the masked images as inputs.

3.9 Generating Output Images

After degrading the images to the proper resolutions, the next step was to fuse the images using both the unmix/sharpen and the sharpen(fuse)/unmix methods.

3.9.1 Fusion

Computer routines were written to implement both the simple ratio and the global regression techniques. The algorithms were tested on simple 9 x 9 and 27 x 27 “images” to verify proper operation. As stated previously, the main problem of fusion occurs with poorly correlated data. Several panchromatic bands were tested. Since the data in all the images ranged from the VIS to the SWIR, there were essentially three choices of panchromatic band. A visible panchromatic image provided good performance in the VIS region, but poor performance in the NIR and SWIR. Panchromatic images created in the NIR and SWIR regions exhibited similar performance: good results were obtained in bands within and adjacent to these areas, with poorer results obtained in the remaining bands.

The main difficulty is that real world objects can exhibit significantly different characteristics in the VIS, NIR, and SWIR. It is difficult to create a fusion engine which adequately performs in all regions. This was one of the main conclusions reached by Braun (1992). The fusion routines created for this work attempted the best performance, with the knowledge that the results would not be perfect.

Since both simple ratio and global regression were implemented, one of Braun’s observations was repeated and is briefly discussed here.

Recall that the simple ratio method performs poorly in areas that are not well correlated. The global ratio method compensates for this but at the price of image resolution. Figure 26 shows fusion of the forest scene using both the simple ratio and global regression methods. The simple ratio output is shown in the pair of

images on the left. The lighter image is a band that is highly correlated with the panchromatic image (band 10). The darker image is a poorly correlated band (band 15). The global regression output is shown in the pair on the right. Band 10 is identical in both image methods, but note that band 10 for the simple ratio image is sharper and less blocky. Apparently, the simple ratio image possesses high frequency information that is not radiometrically correct. The global regression image is radiometrically accurate, but disappointingly blurry. However, since unmixing requires spectral precision, the global regression was used for this research. The fusion was performed using correlation thresholds of 0.85 and 0.90.

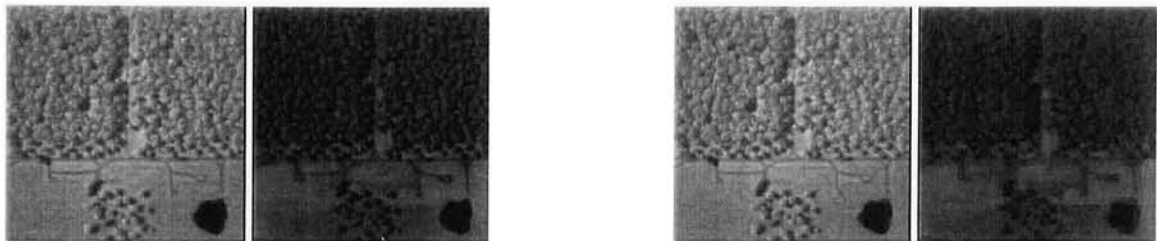


Figure 26: Comparison of Output from Simple Ratio and Global Regression Methods

The program flow for fusion is shown in Figure 27. The header file contains all the useful data (filenames, thresholds, etc.) used to perform fusion. See Figure 28 for a sample header file.

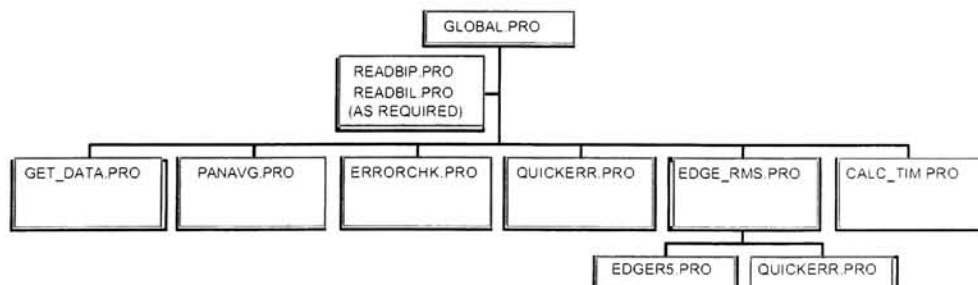


Figure 27: Fusion Program Flow Diagram

The global regression routine proceeds as follows:

1. Read “header” information. Program: get_data.pro.
2. Assign initial variables, read in low-resolution multispectral (LRXS), high-resolution panchromatic (HRP), and LRXS class map images, and calculate averaged panchromatic image. Program: panavg.pro. (readbil.pro, and readbip.pro used as required when reading images that are band interleaved by line -- BIL, or by pixel -- BIP)
3. Determine which bands are well correlated and perform ratio method for those bands.
4. For remaining bands, perform a first order approximation for the band of the form

$$LRXS(n) = a_0 + a_1 \overline{HRP}_s + a_2 LRXS(i) + \varepsilon_i \quad \text{Eq. 57}$$

where n is the band to be predicted, LRXS(i) is a previously predicted band, \overline{HRP}_s is the high-resolution panchromatic image averaged over a superpixel, and a_0 through a_2 are coefficients of the linear regression. The band (LRXS(i)) producing the lowest summed squared error is the best predictive band. Next, the predicted band obtained from equation 57 is correlated with the target band. If the correlation is below the user-determined threshold then go to step 5. If the correlation is above the threshold, then the coefficients are obtained for each class and the global regression is performed using

$$HRXS(n) = a_0 + a_1 HRP + a_2 HRXS(i) \quad \text{Eq. 58}$$

where HRXS(i) is a previously predicted high-resolution band. Go to step 6.

5. Two bands are required for the solution. Perform regression for each class to obtain the coefficients, and perform the high-resolution prediction using

$$HRXS(n) = a_0 + a_1 HRP + a_2 HRXS(i) + a_3 HRXS(j) \quad \text{Eq. 59}$$

where HRXS(j) is a second previously predicted band.

6. Proceed to the next un-predicted band. Repeat steps 4 and 5 until all HRXS bands are predicted.
7. Calculate the effective RMS error. Program: quickerr.pro
8. Calculate the effective edge RMS error. Program: edge_rms.pro
9. Calculate elapsed time. Program: calc_tim.pro
10. End of program.

```

LRXS_filename= c:\rsi\idl40\thesis\western\wrb_06.bsq
pan_filename= c:\rsi\idl40\thesis\western\wrb_p03.bsq
LRXS_class_filename= c:\rsi\idl40\thesis\western\wr_06c20.bsq
TRUTH_filename= c:\rsi\idl40\thesis\western\wrb_03.bsq
output_image= c:\rsi\idl40\thesis\output\fuse_out\western\wf0603.bsq
output_data= c:\rsi\idl40\thesis\output\fuse_out\western\wf0603.dat
LRXS_cols = 156
LRXS_rows= 116
no_bands= 10
mag= 2
no_classes= 20
corr_threshold= 0.940
print_to_screen= n
make_edge_rms= y
LRXS_filetype= BSQ
pan_filetype= BSQ
class_filetype= BSQ
Long_log= n

```

Figure 28: Sample Fusion Header

3.9.2 Unmixing

Computer routines were written to perform stepwise unmixing as described by Gross (1996) and traditional unmixing. The strength of stepwise unmixing is its ability to search through a large database for the optimal endmembers to be unmixed. However, in order to directly compare the two unmixing methods, the spectral libraries used were small (containing fewer endmembers than the number of bands).

The stepwise unmixing routines written by Gross (1996) in MATLAB were used as a starting point. Some of the essential components of the unmixing routines (least squares inequality, non-negative least squares,

etc.) were written and tested during the summer of 1996 by Daisei Konno as part of a funded research project performed by members of the DIRS lab at RIT. The remaining portions (stepwise regression routines) were “translated” from MATLAB to IDL[®]. MATLAB is an ideal application when high mathematical precision is desired. However, its performance with spectral images is awkward. This provided the incentive to transfer the routines to IDL[®], which is more suited to (spectral) image processing.

To validate the IDL[®] code, an image used by Gross (1996) served as input for both the IDL[®] and the MATLAB routines. The code was deemed to be validated when the IDL[®] code could duplicate the output of the MATLAB routines written by Gross.

The program flow for stepwise unmixing is shown in Figure 29. The program flow for traditional unmixing is shown in Figure 30. The header file contains all the useful data (filenames, thresholds, etc.) used to perform unmixing. See Figure 31 for a sample header file.

The material fractions were scaled from 0 to 255, with fractions less than -0.05 given the value of 253, fractions greater than 1.05 given the value of 255, and fractions between -0.05 and 1.05 scaled from 0 to 250. A tiled image similar to the multiple MRI images used by a doctor was also generated, which placed the material maps in order, left to right, and top to bottom (materials were in the same order as given in the library file). A fractions file was generated, containing the raw, unscaled fractions from unmixing to be used in the sharpening routine.

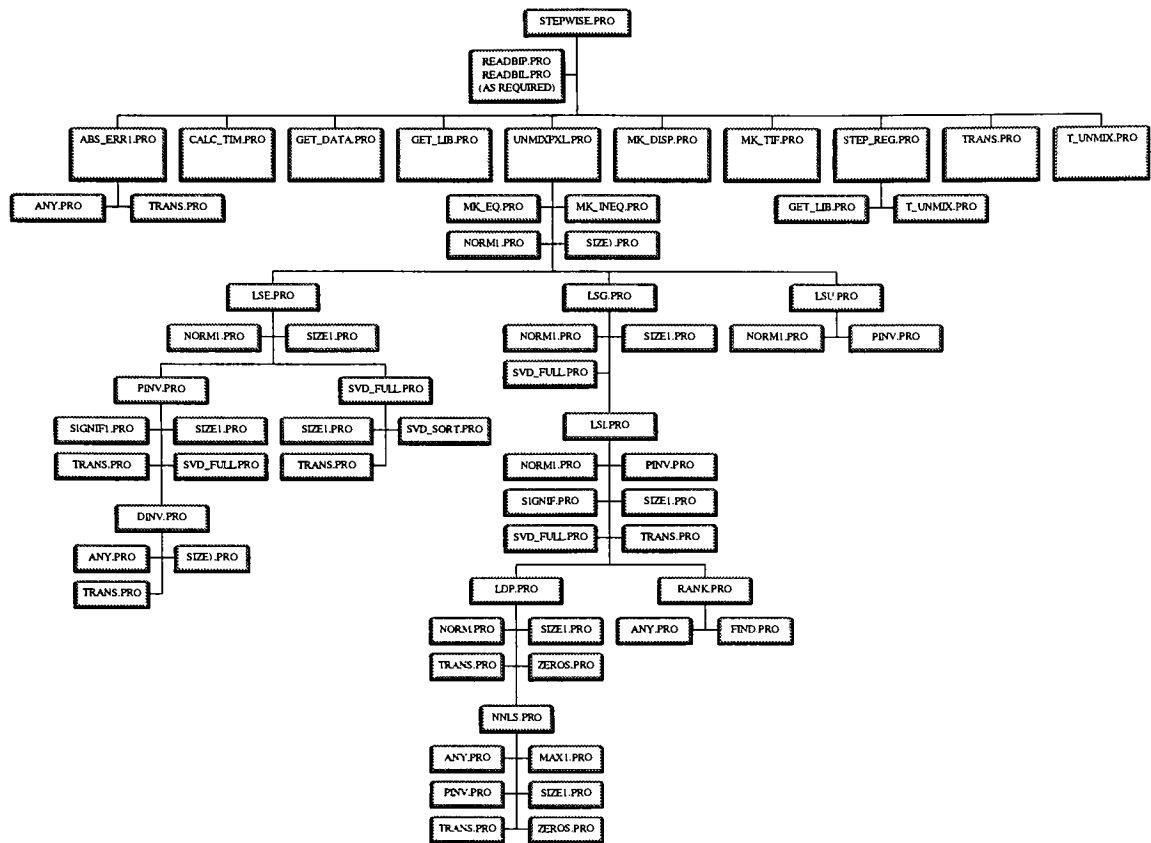


Figure 29: Stepwise Unmixing Program Flow Diagram

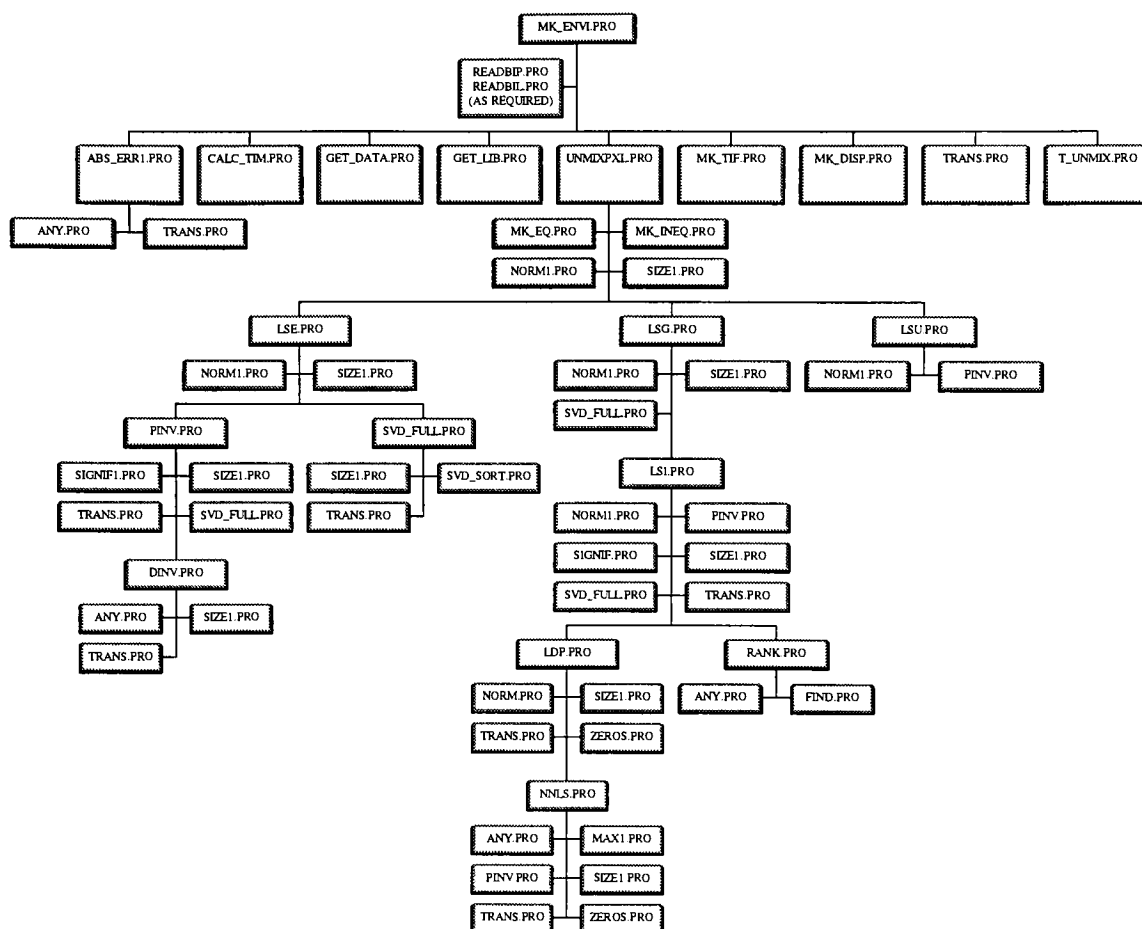


Figure 30: Traditional Unmixing Program Flow Diagram

The stepwise unmixing routine proceeds as follows:

1. Read "header" information. Program: get_data.pro.
2. Assign initial variables and read in low-resolution multispectral image (LRXS), and reflectance library. (readbp.pro and readbil.pro used as required). The algorithm may be performed in reflectance space (0 to 1) or Digital Count (DC) space (0 to 255). Both the multispectral image and the reflectance library must be in the same space. The routine was more stable when

calculations were performed in digital count space, so the spectral library was multiplied by 255 to obtain the desired transformation.

```
INPUT_FILENAME= c:\rsi\idl40\thesis\western\wrb_03.bsq
LIBRARY_FILENAME= c:\rsi\idl40\thesis\western\wrb_lib.dat
REFLECTION_DATA= y
TRUTH_FILENAME= c:\rsi\idl40\thesis\western\dum.dat
MATERIALS_FILENAME= c:\rsi\idl40\thesis\output\unmix_out\western\wu03.bsq
LOG_FILENAME= c:\rsi\idl40\thesis\output\unmix_out\western\wu03.dat
LONG_LOG= n
WIDTH= 312
HEIGHT= 232
NUM_BANDS= 10
NUM_ENDMEMBERS= 8
1 2 3 2 4 5 6 7 8;LIBRARY INDEX
WINDOW_SIZE= 3
UNMIX_CONSTRAINTS= 2
F_ENTER_EXIT= 4.0
PRINT_SCREEN= n
MAKE_FRACTIONS= y
INPUT_FILE_TYPE= BSQ
TRUTH_FILE_TYPE= BSQ
TRUTH_EXISTS= n
```

Figure 31: Sample Unmixing Header

3. For each pixel, obtain LRXS spectral vector and determine optimum endmembers for final (fully constrained) unmixing. Program: step_reg.pro. The step_reg.pro program is the heart of the stepwise method and its operation is now described in detail.

The stepwise regression routine uses a sequential F-test to add and remove endmembers from the model. It contains an outer loop which controls the overall stepwise routine. Within the outer loop is a loop to determine if a variable should be added to the model, and another loop to determine if a variable should be removed from the model.

The add-a-variable loop retains the existing model, and forms candidate supermodels, where each supermodel is formed by adding one of the unused variables to the existing model. The total Sum of Squares is calculated ($SS_{total} = y'y$, where y is the vector of digital counts for the pixel under

investigation). The Sum of Squares for each supermodel is calculated ($SS_{\text{supermodel}} = z'Z'y$, where z is the vector of fractions for the endmembers, and Z is a matrix containing the reflectance values for the endmembers). The fractions in z are obtained via single value decomposition. Note that for the first iteration, the value of z is $[1]$. The Sum of Squares due to the additional variable, $SS_{\text{extra_term}}$, Mean Square due to the extra term, $MS_{\text{extra_term}}$, Sum of Squares due to the residuals, SS_{residual} , and Mean Square due to the residuals, MS_{residual} , are calculated for each supermodel as follows:

$$SS_{\text{extra_term}} = SS_{\text{supermodel}} - SS_{\text{model}} \quad \text{Eq. 60}$$

$$MS_{\text{extra_term}} = SS_{\text{extra_term}}/1 \quad \text{Eq. 61}$$

$$SS_{\text{residual}} = SS_{\text{total}} - SS_{\text{supermodel}} \quad \text{Eq. 62}$$

$$MS_{\text{residual}} = SS_{\text{residual}}/(k-n-1) \quad \text{Eq. 63}$$

where k is the number of bands in the image, and n is the number of terms in the model. Note that for the first iteration, $n = 1$ and $SS_{\text{model}} = 0$. The value of F-to-enter is determined for each supermodel by $F_{\text{enter}} = MS_{\text{extra_term}}/MS_{\text{residual}}$. The supermodel which has the largest F-to-enter ratio (the model which best explains the variance) is then examined further. If the value of F-to-enter is above a user-selected threshold, then the additional variable is significant and added to the model. Since the model has been changed, the Sum of Squares for the model is changed by

$$SS_{\text{model}} = b'X'y \quad \text{Eq. 64}$$

where b is the vector of fractions in the new model ($b = [1]$ after the first iteration), and X is a matrix of reflectance values for the endmembers in b .

The remove-a-variable loop retains the existing model, and forms candidate submodels, where each submodel is formed by removing one of the variables in the existing model. The Sum of Squares for each submodel is calculated ($SS_{\text{submodel}} = z'Z'y$, where z is the vector of fractions for the endmembers in the submodel, and Z is a matrix containing the reflectance values for the

endmembers). The fractions in z are obtained via single value decomposition. The Sum of Squares due to the additional variable, $SS_{\text{extra_term}}$, Mean Square due to the extra term, $MS_{\text{extra_term}}$, Sum of Squares due to the residuals, SS_{residual} , and Mean Square due to the residuals, MS_{residual} , are calculated for each submodel as follows

$$SS_{\text{extra_term}} = SS_{\text{model}} - z'Z'y \quad \text{Eq. 65}$$

$$MS_{\text{extra_term}} = SS_{\text{extra_term}}/1 \quad \text{Eq. 66}$$

$$SS_{\text{residual}} = SS_{\text{total}} - SS_{\text{model}} \quad \text{Eq. 67}$$

$$MS_{\text{residual}} = SS_{\text{residual}}/(k-n-1) \quad \text{Eq. 68}$$

The value of F-to-remove is calculated for each submodel by $F_{\text{remove}} = MS_{\text{extra_term}}/MS_{\text{residual}}$. If the smallest of these F_{remove} values is below the user-selected threshold, then the corresponding endmember is removed from the model. The Sum of Squares for the model is updated using equation 64.

To summarize, first all models with one-endmember models are examined, and the one with the highest F-to-enter value is retained. Next, all two-endmember supermodels containing the previously selected endmember are examined. Once again, the model with the largest F-to-enter value is examined. If the variable is added, then the add-a variable loop will execute again. Once three variables are present in the model, then a variable may be removed by the remove-a-variable loop. This process continues (controlled by the outer loop) until a variable is neither added nor removed.

4. Perform final unmixing (Fully Constrained). Program: unmixpxl.pro.

For example, if stepwise selection returns three endmembers, and there are m bands, the results are $LRXS = Rf$

$$\begin{bmatrix} LRXS_1 \\ LRXS_2 \\ \vdots \\ LRXS_m \end{bmatrix} = \begin{bmatrix} R_{1,1} & R_{1,2} & R_{1,3} \\ R_{2,1} & R_{2,2} & R_{2,3} \\ \vdots & \vdots & \vdots \\ R_{m,1} & R_{m,2} & R_{m,3} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} \quad \text{Eq. 69}$$

where LRXS is a vector of digital counts of the multispectral image, \mathbf{R} is the matrix of the reflectance values for the three materials, and \mathbf{f} is the vector of fractions for the three materials.

The equality constraints are

$$\mathbf{d} = \mathbf{C}\mathbf{f}$$

$$[1] = [111] \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} \quad \text{Eq. 70}$$

and the inequality constraints are

$$\mathbf{h} \leq \mathbf{G}\mathbf{f}$$

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ -1 \\ -1 \\ -1 \end{bmatrix} \leq \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} \quad \text{Eq. 71}$$

The equality and inequality problems are solved as described in Appendices A and B.

5. Proceed to next pixel, repeating steps 3 - 4.
6. If truth maps are available, (when working with DIRSIG images) then calculate squared error.
Program: abs-err1.pro.
7. Make displayable byte image by scaling fraction maps to digital count space. Program:
mk_disp.pro.

8. Make “tiled” output image similar to the multiple MRI (Magnetic Resonance Imaging) images used by a doctor. Program: `mk_tif.pro`
9. Calculate elapsed time. Program: `calc_tim.pro`
10. End of program.

The traditional unmixing routine proceeds as follows:

1. Read “header” information. Program: `get_data.pro`.
2. Assign initial variables and read in LRXS, and reflectance library. (`readbip.pro` and `readbil.pro` used as required).
3. For each pixel, perform final unmixing (Fully Constrained). Program: `unmixpxl.pro`. The example used in describing the final unmixing for the stepwise routine is applicable for traditional unmixing also. The only difference is that the number of endmembers to be unmixed is equal to the number of materials in the reference library, and is the same for every pixel.
4. Proceed to next pixel, repeating step 3.
5. If truth maps are available, (when working with DIRSIG images) then calculate squared error. Program: `abs_err1.pro`.
6. Make displayable byte image by scaling fraction maps to digital count space. Program: `mk_disp.pro`.
7. Make “tiled” output image. Program: `mk_tif.pro`
8. Calculate elapsed time. Program: `calc_tim.pro`
9. End of program.

3.9.3 Sharpening

Computer routines were written to perform sharpening as described by Gross (1996). As with unmixing, the MATLAB sharpening routines were used as a starting point.

Both partially constrained and fully constrained algorithms were written in IDL[®]. However, the fully constrained code was not finalized and validated. Gross states that the improvement obtained with fully constrained unmixing is not really worth the computational complexity and longer run times. For example, a simple (minimum texture involved) DIRSIG forest scene was unmixed at 16X and sharpened to 2X. When sharpened using partial constraints, the squared error was 0.3706 and the program required ten hours, 37 minutes to complete. The fully constrained algorithm had a squared error of 0.3630 and took 98 hours, 10 minutes! The slightly improved error did not merit increasing the run time by nearly a factor of ten.

The partially constrained sharpening algorithm (in IDL[®]) was validated using the existing MATLAB code. When both routines generated identical output (from identical inputs) the IDL[®] code was deemed to be validated.

The program flow for sharpening is shown in Figure 32. The header file contains all the useful data (filenames, thresholds, etc.) used to perform sharpening . See Figure 33 for a sample header file.

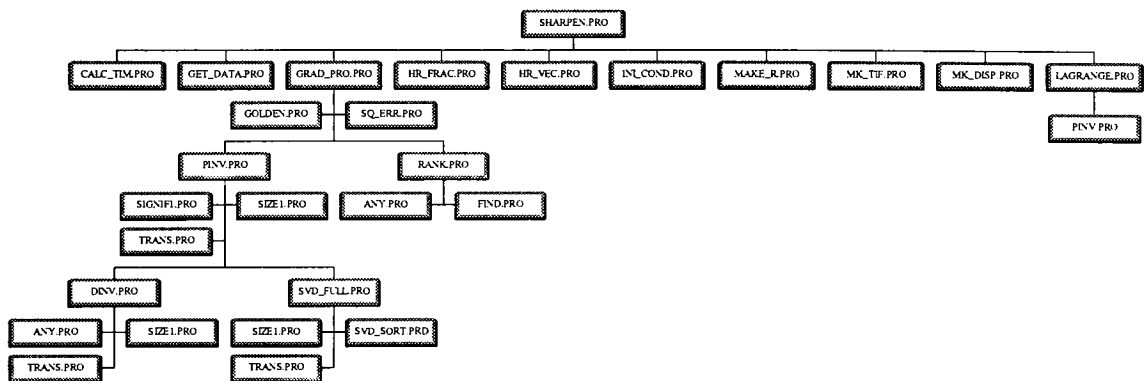


Figure 32: Sharpening Program Flow Diagram

The material fractions were scaled from 0 to 255, with fractions less than -0.05 given the value of 253, fractions greater than 1.05 given the value of 255, and fractions between -0.05 and 1.05 scaled from 0 to 250. A

tilled image similar to the multiple MRI images used by a doctor was also generated, which placed the material maps in order, left to right, and top to bottom (materials were in the same order as given in the library file).

```
LR_MAT_FILENAME= c:\rsi\idl40\thesis\output\unmix_out\western\wu24f.bsq
LIBRARY_FILENAME= c:\rsi\idl40\thesis\western\wrb_plib.dat
REFLECTION_DATA= y
TRUTH_FILENAME= c:\rsi\idl40\thesis\western\dum.dat
MATERIALS_FILENAME= c:\rsi\idl40\thesis\output\sharpen_out\western\ws2412.bsq
LOG_FILENAME= c:\rsi\idl40\thesis\output\sharpen_out\western\ws2412.dat
LONG_LOG= n
HI_RES_FILENAME= c:\rsi\idl40\thesis\western\wrb_p12.bsq
WIDTH= 39
HEIGHT= 29
HR_NUM_BANDS= 1
NUM_ENDMEMBERS= 8
scale= 2
CONSTRAINTS= 1
NUM_SHARPENING_BANDS= 1
0 ;SHARPENING BANDS
PRINT_SCREEN= n
TRUTH_EXISTS= n
```

Figure 33: Sample Sharpening Header

The sharpening routine proceeds as follows:

1. Read “header” information. Program: get_data.pro.
2. Assign initial variables and read in low-resolution fraction maps, HRP, and reflectance library.

Program: hr_vec.pro (readbil.pro and readbip.pro as required). Sharpening may also be performed in either reflectance space or digital count space. There was no noted instability of the routines in digital count space vs reflectance space as was observed in unmixing, so sharpening was performed in reflectance space. The results are the same, and it is really “programmer’s choice” of which space to use in performing calculations.

3. For each pixel, obtain low-resolution fractions and corresponding digital counts from HRP.

4. Create initial conditions and constraint vectors/matrices. Program: ini_cond.pro. For example, at a sharpening scale factor of 2X for three endmembers, the result is

$$HRP = R_P \tilde{f}$$

$$\begin{bmatrix} HRP_1 \\ HRP_2 \\ HRP_3 \\ HRP_4 \end{bmatrix} = \begin{bmatrix} R_{P,1} & R_{P,2} & R_{P,3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & R_{P,1} & R_{P,2} & R_{P,3} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & R_{P,1} & R_{P,2} & R_{P,3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & R_{P,1} & R_{P,2} & R_{P,3} \end{bmatrix} \tilde{f} \quad \text{Eq. 72}$$

where HRP is a vector of digital counts in the high-resolution sharpening band, R is the matrix of panchromatic reflectance values for the three materials, and \tilde{f} is a vector of the fractions for the endmembers in the four subpixels in the high-resolution band. This vector in long form is (in order to save space on the page, the transpose will be written)

$$\tilde{f}' = [f_{1,1} \ f_{2,1} \ f_{3,1} : f_{1,2} \ f_{2,2} \ f_{3,2} : f_{1,3} \ f_{2,3} \ f_{3,3} : f_{1,4} \ f_{2,4} \ f_{3,4}] \quad \text{Eq. 73}$$

where the first subscript refers to the endmember, and the second subscript refers to the subpixel location.

The consistency constraints are written as

$$\begin{bmatrix} 4f_1 \\ 4f_2 \\ 4f_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & : & 1 & 0 & 0 & : & 1 & 0 & 0 & : & 1 & 0 & 0 \\ 0 & 1 & 0 & : & 0 & 1 & 0 & : & 0 & 1 & 0 & : & 0 & 1 & 0 \\ 0 & 0 & 1 & : & 0 & 0 & 1 & : & 0 & 0 & 1 & : & 0 & 0 & 1 \end{bmatrix} \tilde{f} \quad \text{Eq. 74}$$

and the equality constraints are

$$\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & : & 0 & 0 & 0 & : & 0 & 0 & 0 & : & 0 & 0 & 0 \\ 0 & 0 & 0 & : & 1 & 1 & 1 & : & 0 & 0 & 0 & : & 0 & 0 & 0 \\ 0 & 0 & 0 & : & 0 & 0 & 0 & : & 1 & 1 & 1 & : & 0 & 0 & 0 \end{bmatrix} \tilde{f} \quad \text{Eq. 75}$$

The inequality constraint vectors and matrices are created also because the program was written during the development of the fully constrained sharpening routine. However, they are not needed for the partially constrained sharpening used for this research.

5. If there is more than one material within superpixel, then there may be room for optimization.

Send all information to the appropriate specific sharpening routine. Program: lagrange.pro.

Lagrange implements equation 47 in the form $AX = B$ where

$$A = \begin{bmatrix} Q & H' \\ H & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} R'_p d \\ h \end{bmatrix}$$

$$X = \begin{bmatrix} x \\ \lambda \end{bmatrix} \quad \text{Eq. 76}$$

where $Q = R'_p R_p + \delta I(n)$ (Note: $I(n)$ is an $n \times n$ identity matrix and δ is a small positive constant. The addition of the small portion along the diagonals prevents the algorithm from becoming “trapped”), H is the matrix of equality constraints, h is the vector of equality constraints, d is a vector containing the digital counts from the sharpening band(s), and x is a vector of the high-resolution fractions to be predicted.. The estimate is performed by

$$X = A^{-1} B \quad \text{Eq. 77}$$

The estimated high-resolution fractions will be the first n elements of the vector X (where n is the number of elements in \tilde{f}).

6. Spatially assign the high-resolution fractions returned by sharpening program to the appropriate places in the high-resolution fraction maps. Program: hr_frac.pro.
7. Proceed to next pixel, repeat steps 3 - 6.
8. If truth fraction maps are available (when working with DIRSIG images) then read in truth fractions, and calculate squared error.
9. Make displayable byte image by scaling fraction maps to digital count space. Program: mk_disp.pro

10. Make “tiled” output image. Program: mk_tif.pro
11. Calculate elapsed time. Program: calc_tim.pro
12. End of program.

4. RESULTS

Use of synthetic imagery allowed for quantitative evaluation of the algorithms discussed. In addition, qualitative evaluations were performed for the real scenes. This section will present quantitative data derived from the synthetic imagery as well as qualitative evaluations of the fraction maps obtained for real and synthetic images.

4.1 PC vs UNIX

The algorithms generated were implemented on both an IBM-Compatible (PC) and a UNIX-based DEC Alpha workstation. The results from both platforms generally agreed to the nearest hundredth. It should be noted, however, that the algorithms ran longer on the PC. In addition, the difference in machine precision was notable between the two machines. Algorithms running on the PC would often crash because they encountered many floating point errors (floating division by zero) that were not encountered on the workstation.

4.2 CPU TIME

The fraction maps generated by the traditional unmix/sharpen method take a very long time to sharpen. It should be expected that each time resolution is doubled, the run time should quadruple (because the number of pixels to be processed increases by a factor of four). Run times for the forest scene are shown in Figure 34. For comparison, the squared errors are plotted along the bottom of the chart (See left-hand axis for numbers), and the run times are plotted along the chart of the top (See right-hand axis for numbers). Although the run times can increase when many users are on the system, the times indicated in the graph are typical of the relative run times for different resolutions. The run times for sharpening the traditionally unmixed fraction maps are very high. This would be acceptable if the squared error was comparable to the other two methods, but it is typically

much larger. The run times for Stepwise Unmix/Sharpen and Fuse/Unmix are comparable, so time should not be a significant factor in choosing one of these two methods over the other.

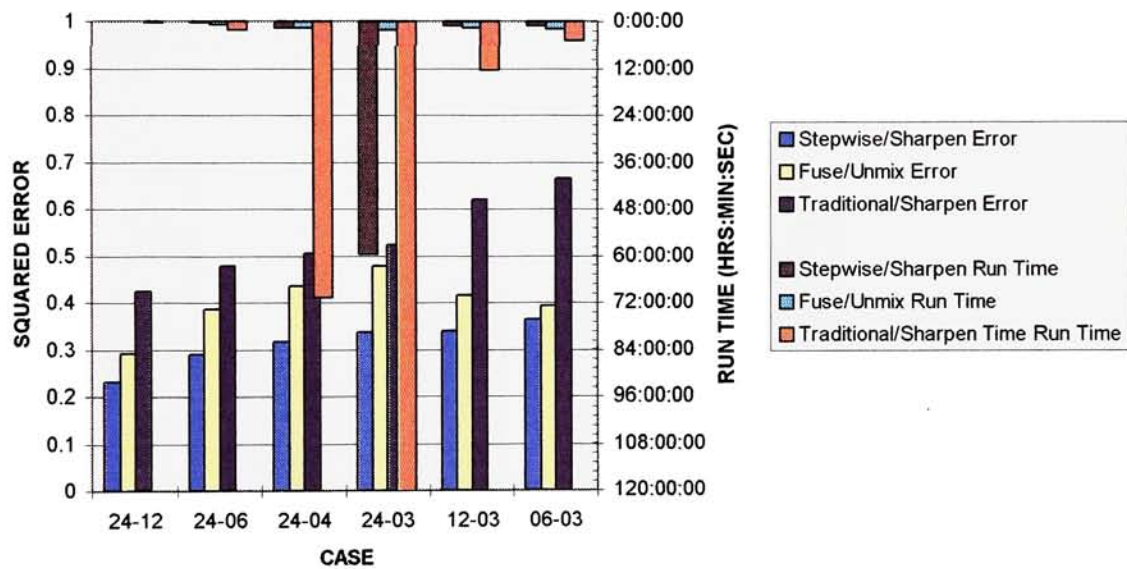


Figure 34: Squared Error and Run Times (Forest Scene)

4.3 RMS Error

The accuracy of the fusion algorithm was determined by examining the effective image-wide RMS and edge RMS numbers. The correlation thresholds used ranged from 0.85 for the Forest scene, to .90 for the Western Rainbow Scenes. Braun (1992) obtained RMS numbers for global regression that ranged from 4.24 to 7.76, and edge RMS numbers from 6.14 to 51.92. Braun also noted that RMS increased with scale factor. RMS

numbers for the forest and desert scenes are shown in Figures 35 and 36. Note that the error increases as the scale factor increases as stated by Braun. The edge RMS is higher than the image-wide RMS, which is not surprising. Recall that the global regression produces softened edges for poorly correlated bands. Braun performed geometric registration as part of his research on image fusion. The effects due to warping the high-resolution PAN image to the low-resolution multispectral image cause an increase in the RMS figures. Registration was not required for the images used in this research, which should explain why the RMS figures obtained are lower than the average RMS numbers obtained by Braun.

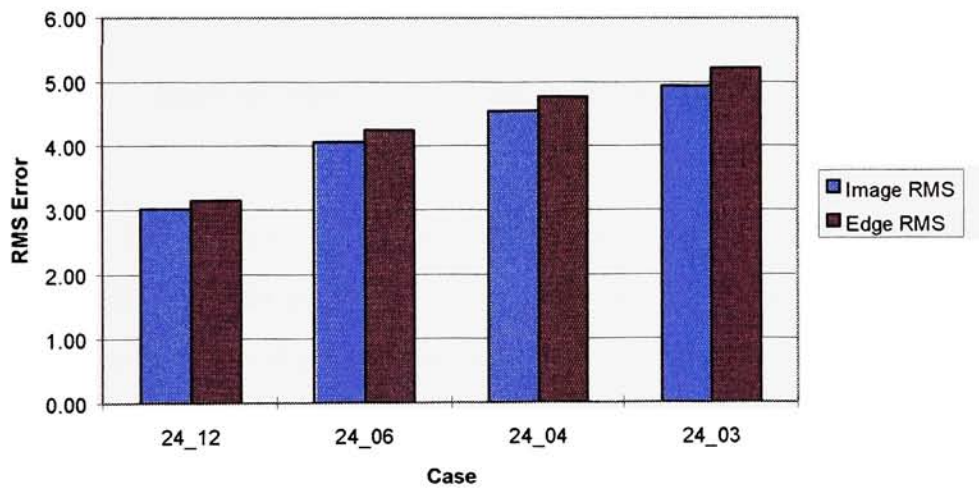


Figure 35: RMS Errors for Fusion of Forest Image

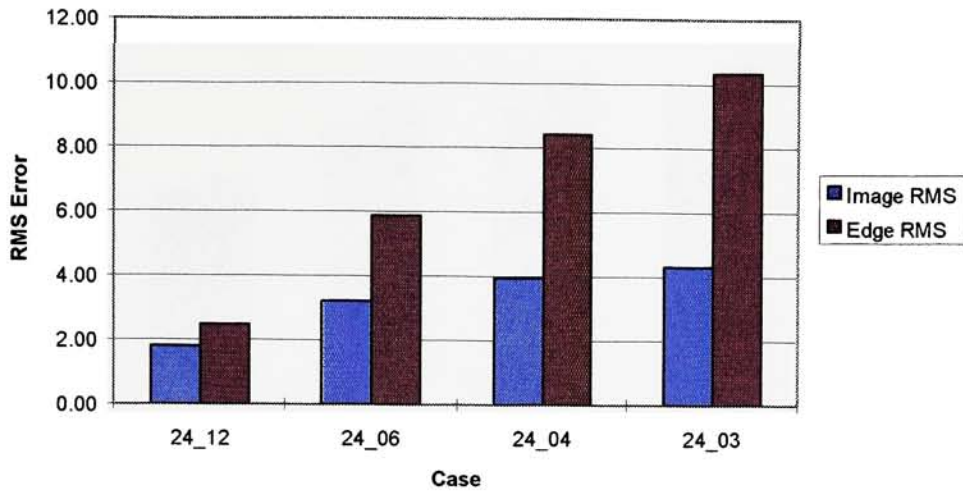


Figure 36: RMS Errors for Fusion of DAEDALUS Image

RMS figures were equal to or less than values obtained by Braun for all but the Rochester scene. The RMS values were significantly higher, as shown in Figure 37. The higher RMS figures are probably due to some of the modeling effects of DIRSIG. The Rochester scene looks synthetic and unreal, compared to the Forest scene. The colors in the Rochester scene seem much too bright and uniform, and “weather” effects are not visible, giving everything a “newly painted” look. This may account for the higher than expected RMS values.

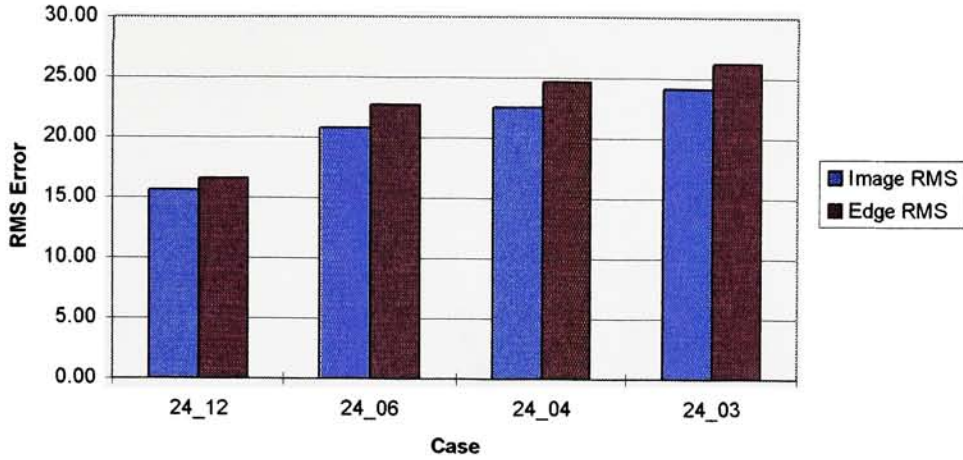


Figure 37: RMS Errors for Fusion of Rochester Image

4.4 Effects of Scale

Both unmixing methods are sensitive to the scale of the images. The squared error increases as the resolution increases. Figures 38 and 39 show the results of unmixing at various scales. Figures 40 and 41 demonstrate the results of the entire process (fuse/unmix and unmix/sharpen) at various scales. The increase in error as resolution increases is not a surprising result. At low resolutions, the pixels are large and contain mixtures of many materials. Most mistakes made are in allocating the (sometimes small) fractions among the elements within the pixel, and the resulting errors are relatively small. As resolution increases, the “purity” of the pixels increases, and an individual pixel is more likely to contain a large amount of one material, and a small amount of one or two others. Consider traditional unmixing, which assigns fractions to all endmembers for each pixel. Forcing a mixture solution (with many elements) in a pixel which does not contain a mixture (or contains a mixture of two or three elements) can result in a fairly large error. Stepwise unmixing is also affected by the increase in purity of the pixels, but since it selects the most likely endmembers, it forces a mixture of fewer

elements than traditional unmixing, and can even select only one endmember, improving its results on completely pure pixels.

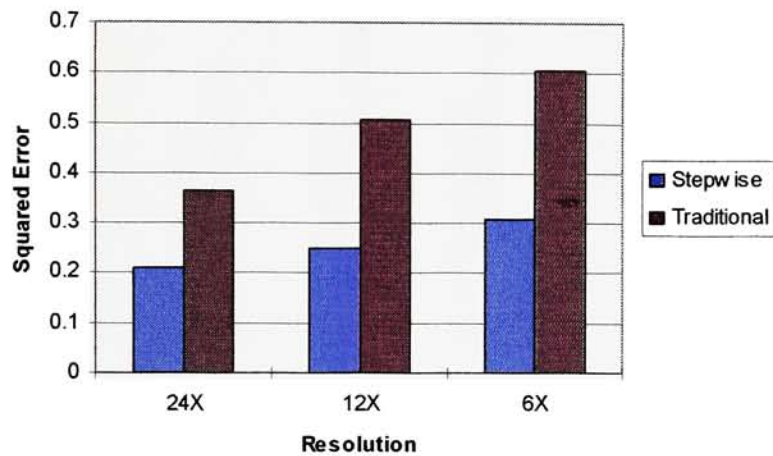


Figure 38: Unmixing Forest Scene at Various Resolutions

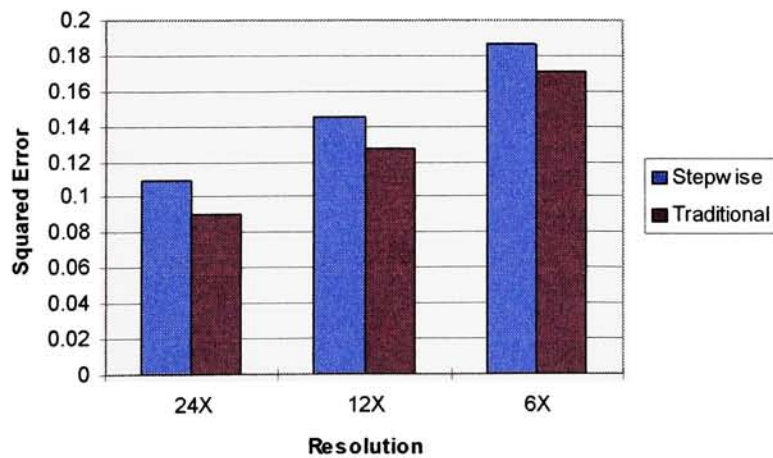


Figure 39: Unmixing Rochester Scene at Various Resolutions

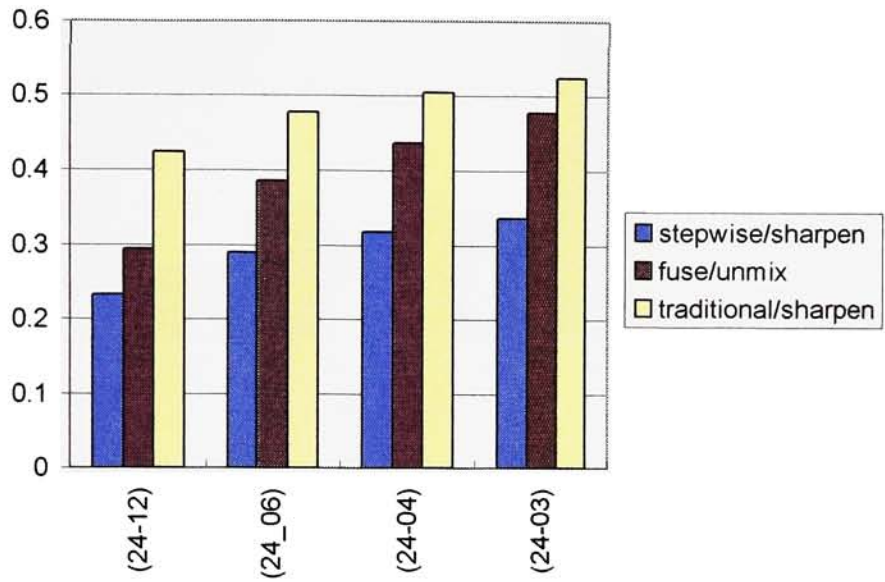


Figure 40: Image Enhancement for Forest Scene at Various Scale Factors

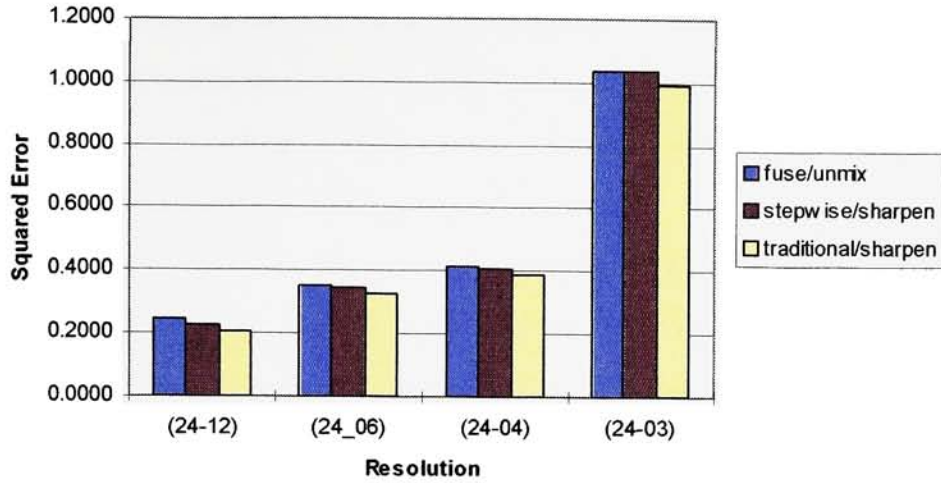


Figure 41: Image Enhancement for Rochester Scene at Various Scale Factors

4.5 Number of Endmembers

The Rochester Scene may not be a good image to use in comparing traditional unmixing to Stepwise unmixing. This scene was generated using 32 materials (several types of wood, brick, and shingle). In order to compare the two unmixing methods, the number of endmembers was required to be less than or equal to 15 (the number of bands in the M-7 Image). This required combining materials with similar spectral curves into one material. The class map was changed appropriately also to contain the smaller number of combined materials.

The unmixing results shown in Figure 41 do not match expectations. The performance of the Stepwise unmixing algorithm is poorer than traditional unmixing for the reduced library. It seems that combining spectral curves as was performed to enable comparison of the two methods degrades the performance of stepwise unmixing (and may improve the performance of traditional unmixing). Recall that the big advantage of stepwise unmixing is that it can be used with a large spectral library. In most cases, the user may not know exactly every material in a scene and may use several materials and let the algorithm determine which ones are present. This

type of operation is not possible using traditional unmixing. As a realistic metric of the “true” performance of the stepwise algorithm, stepwise unmixing was performed using the full 32-material spectral library, resulting in a squared error of 0.1530 (which shows that the unmixing works rather well). The squared error for the 32-material fraction maps cannot really be compared to the error for the 15-material library. A proposed revision to the squared error metric is discussed in the Conclusions Section.

Although the Rochester scene does not allow a good comparison between the unmixing methods, it does highlight one of the limitations of traditional unmixing. The error in traditional unmixing decreases as the number of endmembers increases. Figure 42 compares the results of traditional unmixing of the Rochester scene using 5, 10, and 15 endmembers. The squared error for the 15-endmember case represents the lower limit for error in traditional unmixing. More endmembers cannot be used to decrease the error further because this method is limited by the number of bands in the image. Stepwise unmixing presents an opportunity to overcome this limitation.

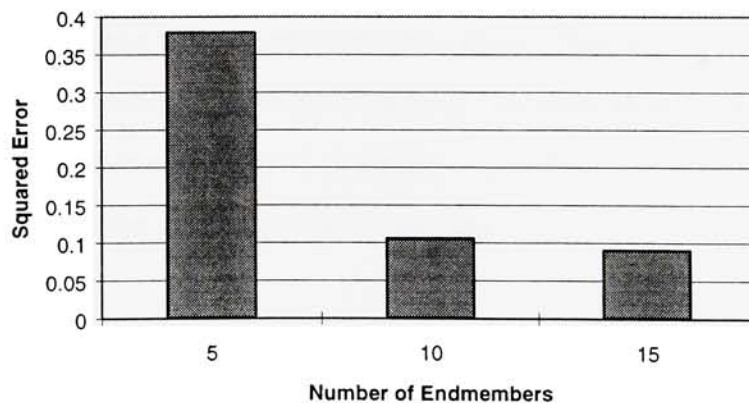


Figure 42: Results of Traditional Unmixing for Rochester Scene with Various Numbers of Endmembers

4.6 Effects of Shadow

Green (1993) states that a shadow endmember can be used to improve results. In most applications, shadow is not retained as a fraction map. The shadow is removed by distributing its fraction equally among the remaining fractions in the target pixel by

$$F_{\text{Shadow}} + \sum F_{\text{Remaining}} = \text{Total}$$
$$F_{\text{Remaining_New}} = F_{\text{Remaining}} * \frac{1}{(\text{Total} - F_{\text{Shadow}})} \quad \text{Eq. 78}$$

where F_{Shadow} is the fraction of shadow in the pixel, $F_{\text{Remaining}}$ are the fractions for the remaining endmembers, and $F_{\text{Remaining_New}}$ are the new fractions for the remaining endmembers. As a numerical example, consider a simple case of a pixel containing shadow, trees, and grass in equal fractions. ($F_{\text{shadow}} = F_{\text{tree}} = F_{\text{grass}} = 1/3$). The new fractions for trees and grass become

$$F_{\text{tree_new}} = F_{\text{grass_new}} = \frac{1}{3} * \frac{1}{(1 - 1/3)} = \frac{1}{2}$$

The forest scene has many shaded areas, providing the opportunity to examine the effect of shadow. This scene was processed without a shadow endmember in the library. These results were compared with those obtained when a shadow endmember was included in the library. The results are shown in Figures 43 through 45, where the shadow has been removed prior to calculating the squared error. Surprisingly use of a shadow endmember had little effect when performing traditional unmixing.

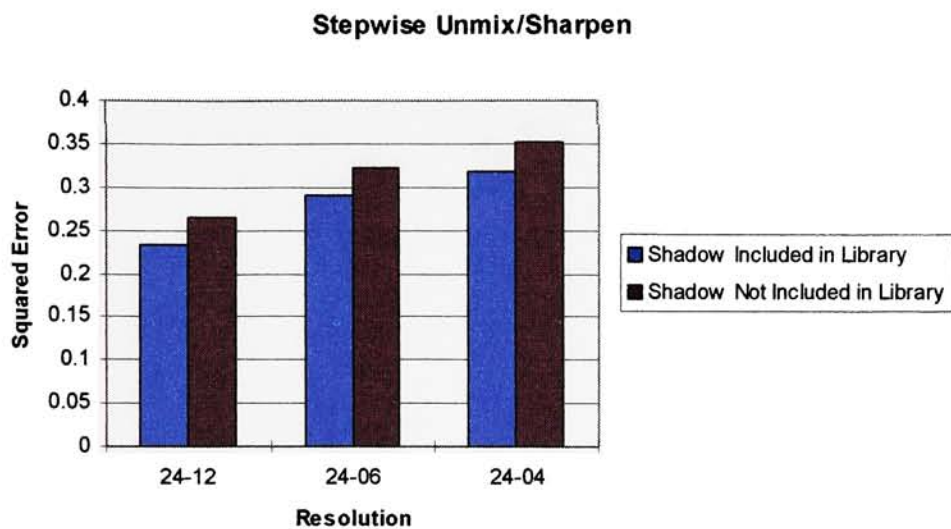


Figure 43: Effect of Shadow Endmember (Stepwise Unmix/Sharpen)

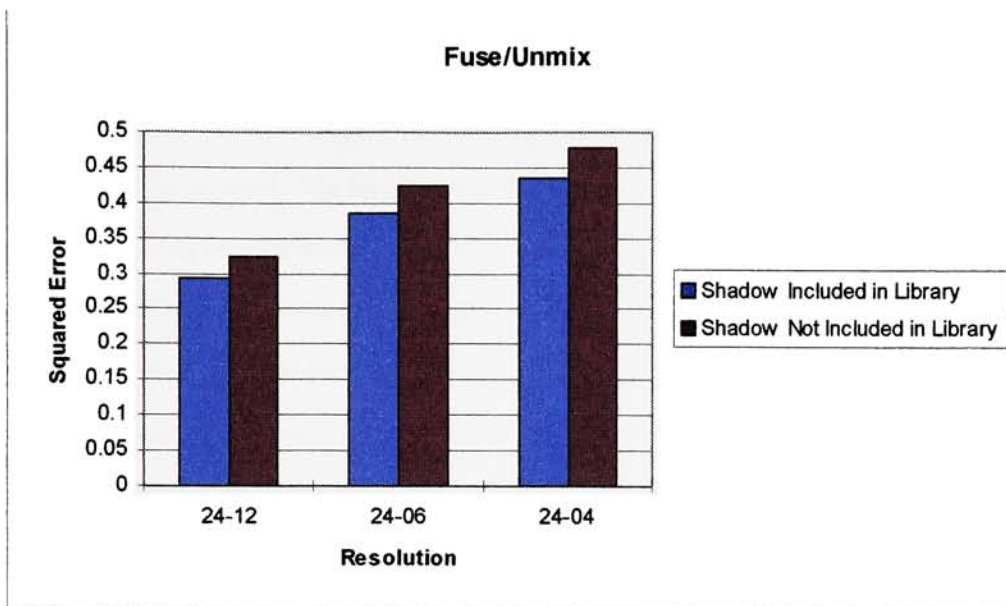


Figure 44: Effect of Shadow Endmember (Fuse/Unmix)

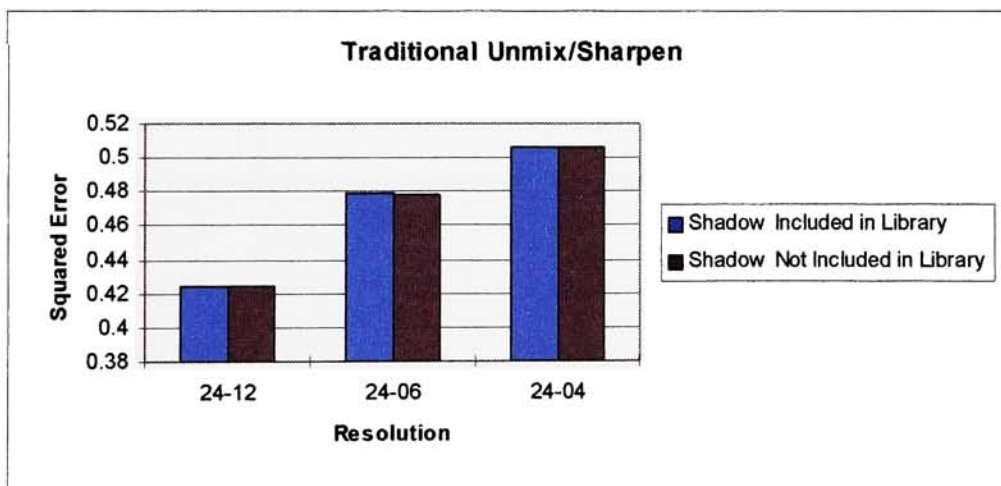


Figure 45: Effect of Shadow Endmember (Traditional Unmix/Sharpen)

The effects of unmixing with the shadow endmember can also be seen by visually examining the fraction maps. Refer to Figures 47 through 52, which show results of the 24-06 case for the Forest scene. The key to the fraction maps is shown in Figure 46. Notice that the stepwise method performs much better when a shadow endmember is included. The fraction maps are “cleaner” and there are fewer stray pixels. So the use of the shadow endmember seems to improve the results of the selection algorithm. Use of a shadow endmember provided less significant improvements in the traditional unmixing method (See quantitative results in Figure 45). As shown in Figures 49 and 50, the visual difference between the results with and without shadow are less pronounced for traditional unmixing than for the methods using stepwise unmixing. In the figures depicting fraction maps, values between -0.05 and 1.05 are displayed as grayscale, values greater than 1.05 are displayed as red, and values less than -0.05 are displayed as green. The occurrences of “out of bounds” fractions are very rare in the unmixed fraction maps, and are generally produced by the sharpening process. The occurrences of negative out of bounds values (green) are more prevalent than those for positive out of bounds values (red). Observe that the soft features generated by the global regression fusion method show up in the fraction maps generated by the fuse/unmix method. For comparison, the truth fractions of the forest scene at a resolution of 4X are shown in Figure 53.

CAMO PAINT	WATER	GRASS	RUBBER	DIRT
TREE BARK	GLASS	TREES	STEEL SIDE	STEEL BUMPER
CANVAS				

Figure 46: Key to Forest Scene Fraction Maps

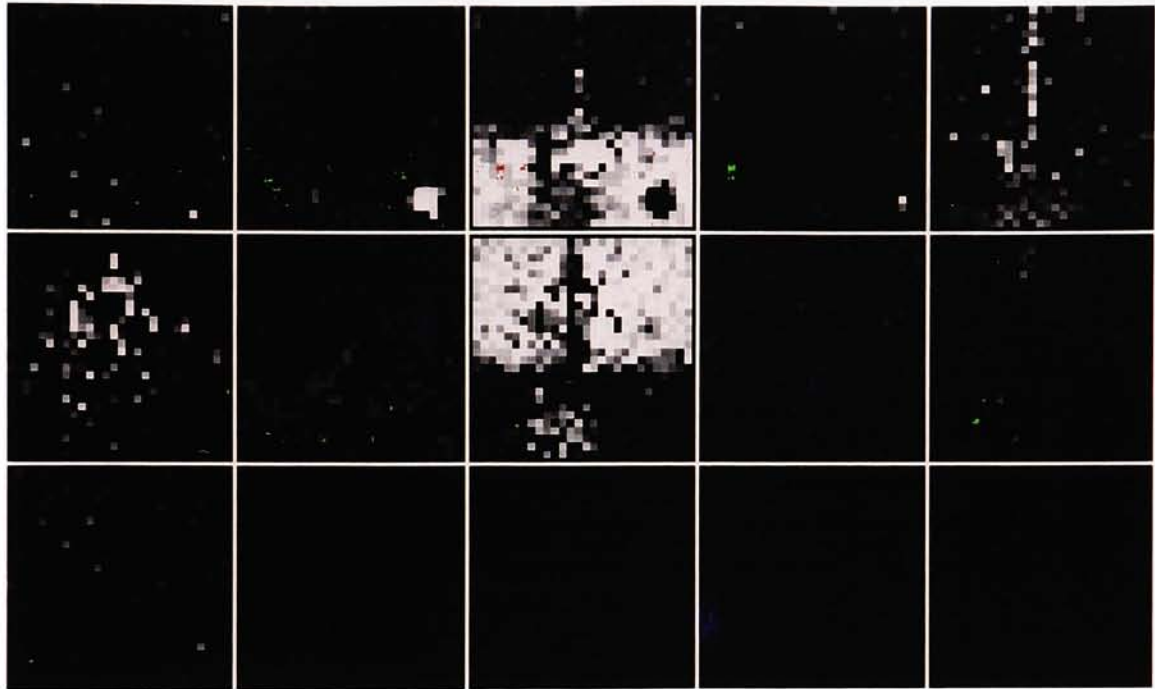


Figure 47: Forest Scene Fraction Maps (Stepwise Unmix/Sharpen) Without Shadow Endmember

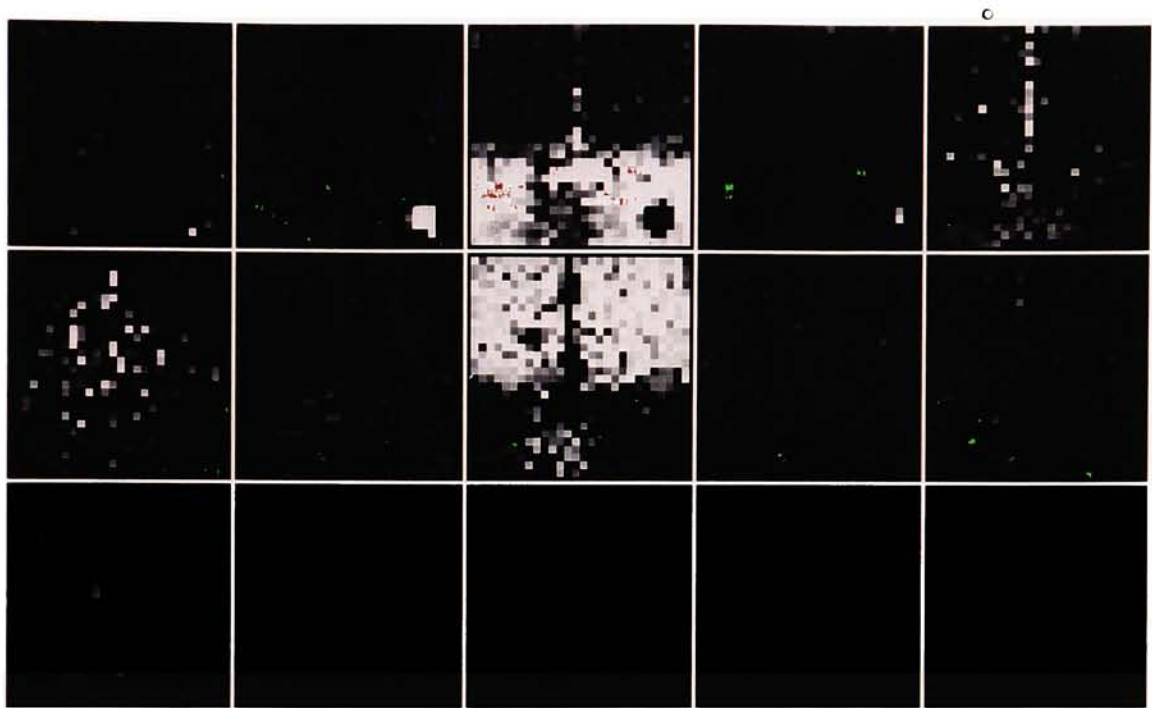


Figure 48: Forest Scene Fraction Maps (Stepwise Unmix/Sharpen) With Shadow Endmember

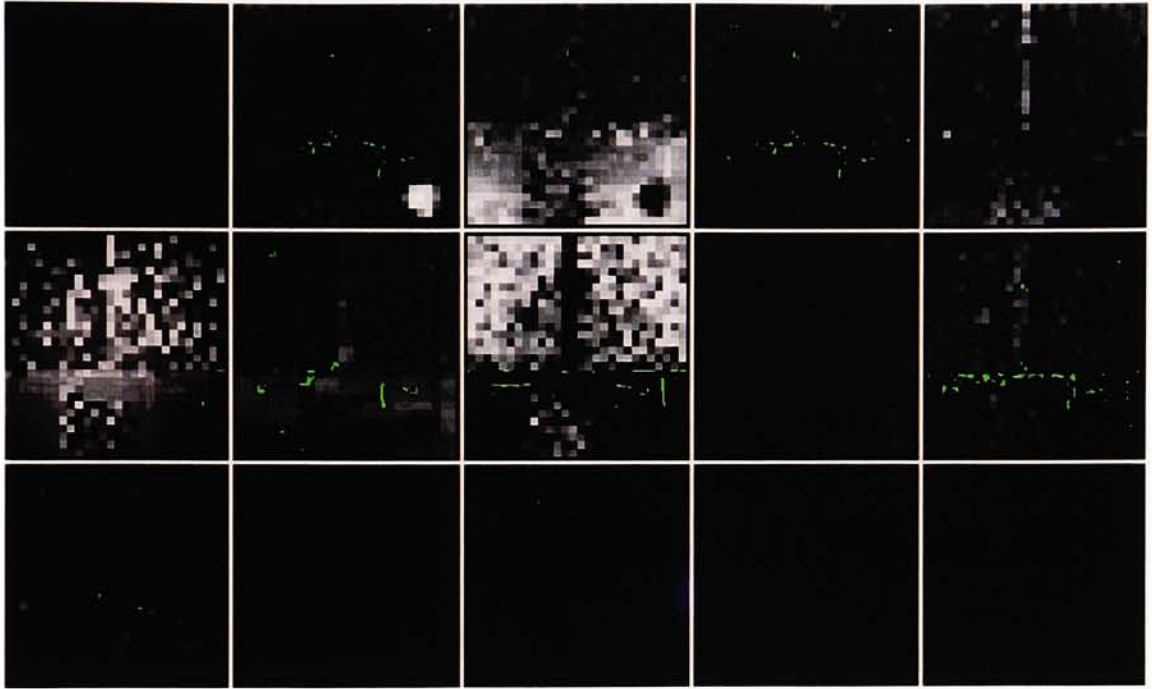


Figure 49: Forest Scene Fraction Maps (Traditional Unmix/Sharpen) Without
Shadow Endmember

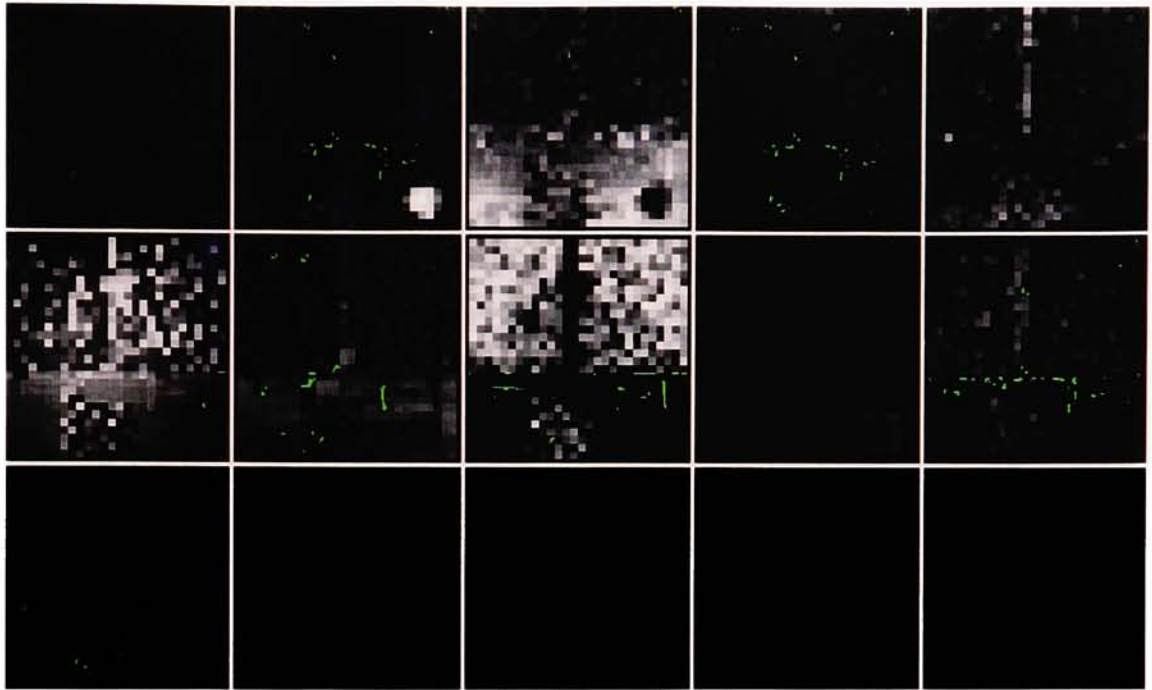


Figure 50: Forest Scene Fraction Maps (Traditional Unmix/Sharpen) With
Shadow Endmember

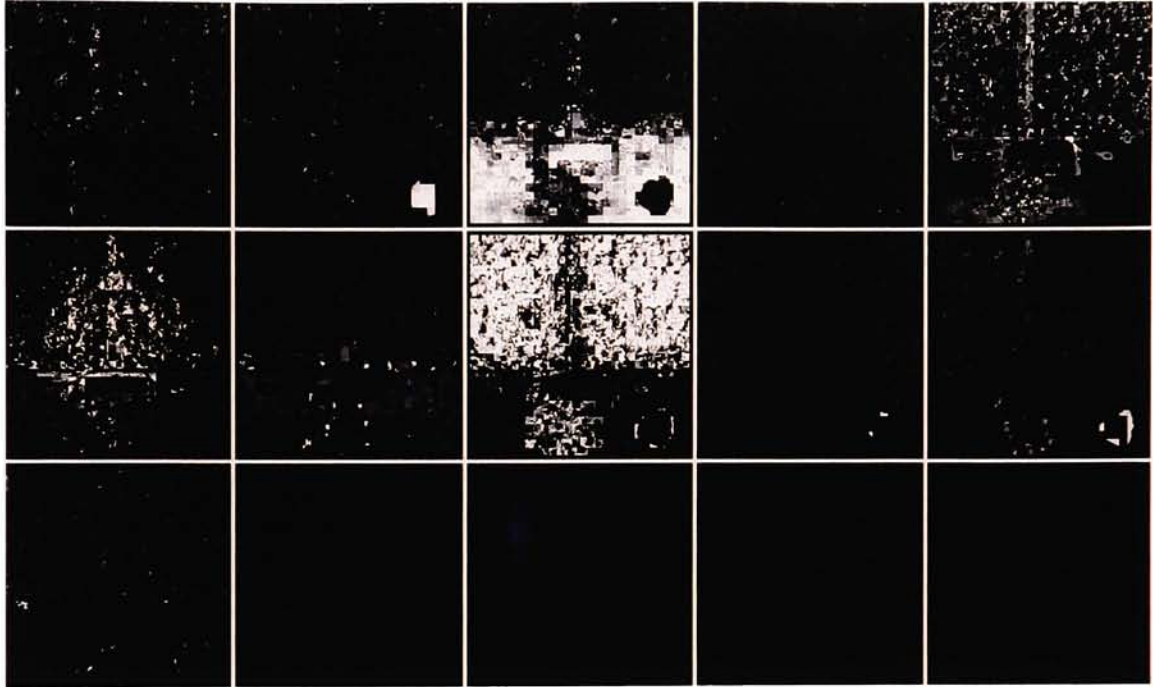


Figure 51: Forest Scene Fraction Maps (Fuse/Unmix Method) Without Shadow Endmember

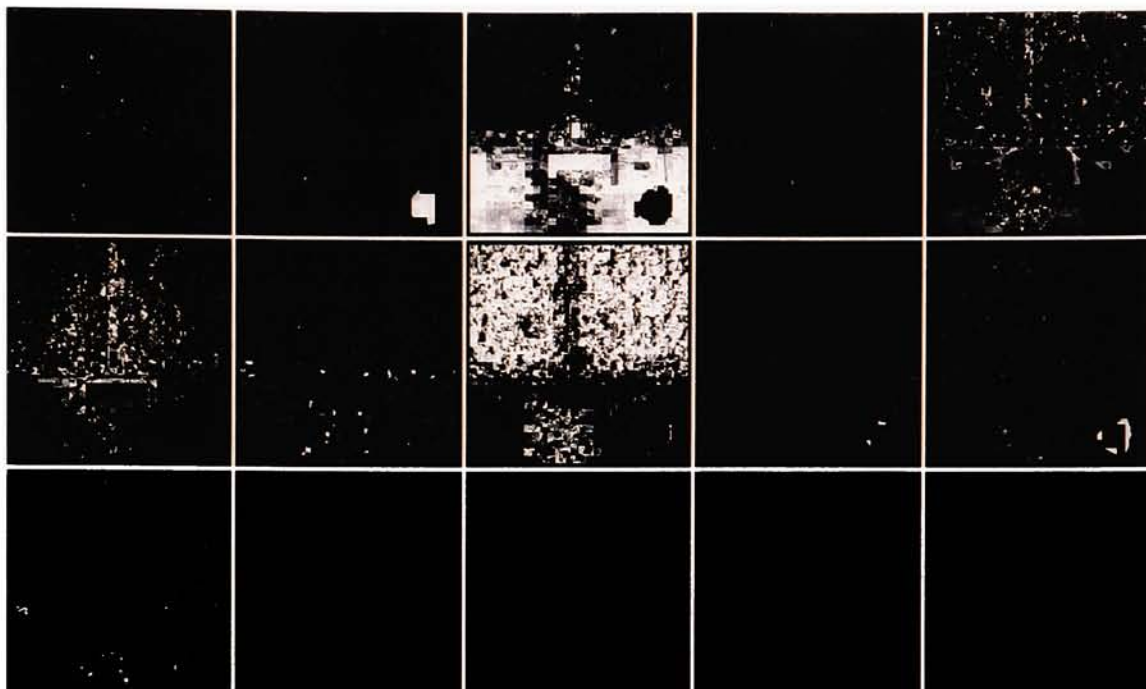
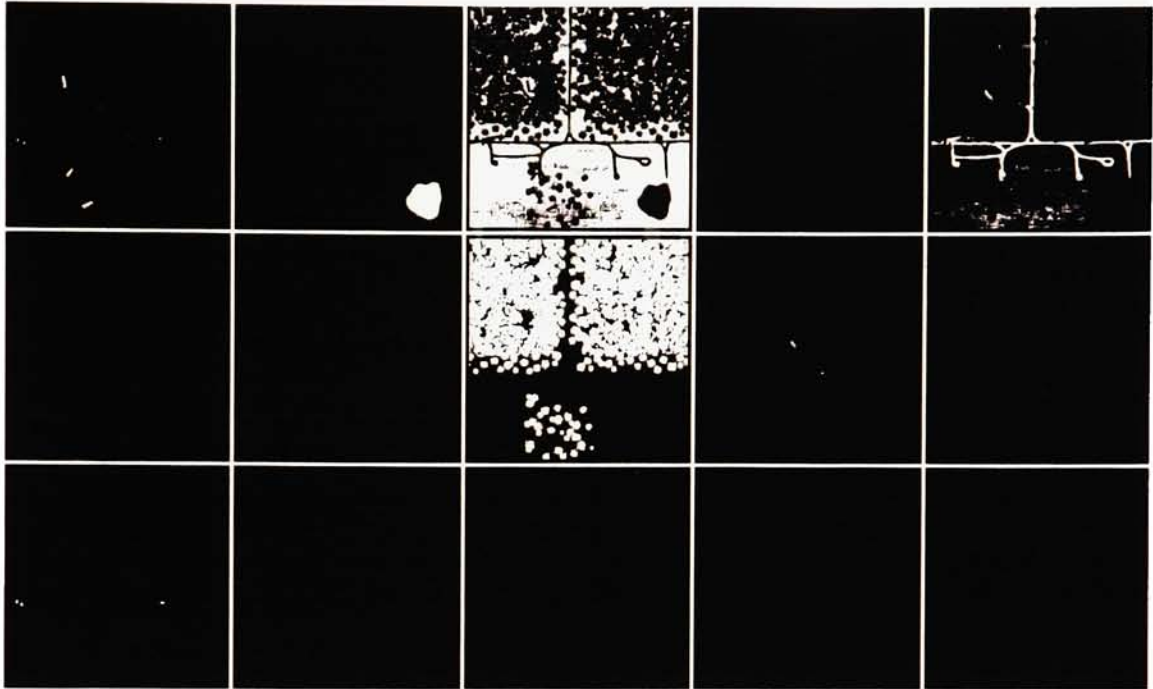


Figure 52: Forest Scene Fraction Maps (Fuse/Unmix Method) With Shadow Endmember



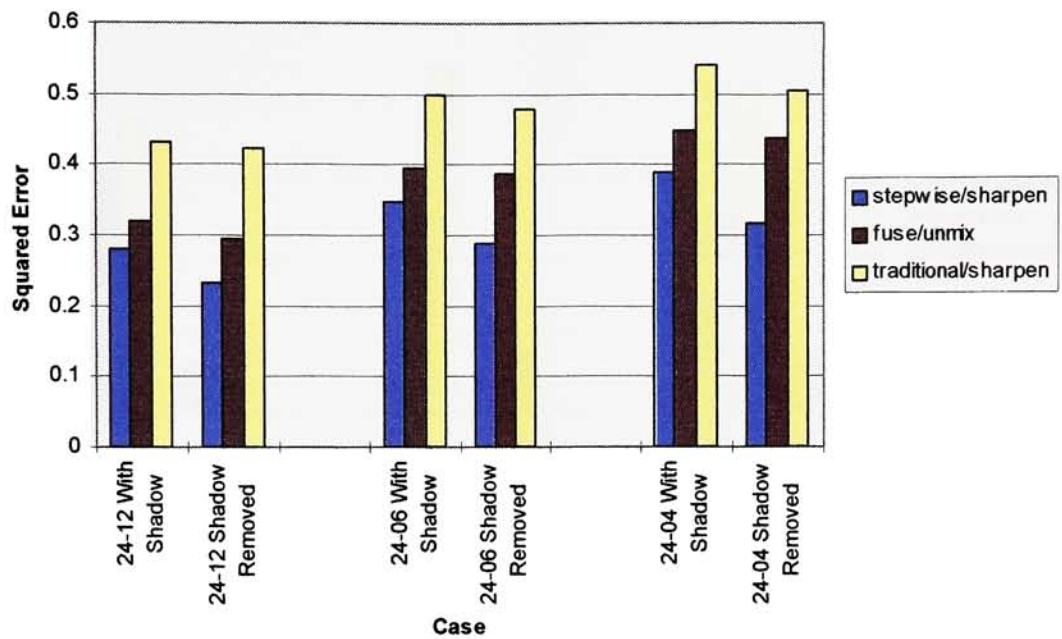


Figure 54: Error Calculated from Shadow Class Map vs Error Calculated from Class Map Without Shadow (one of two)

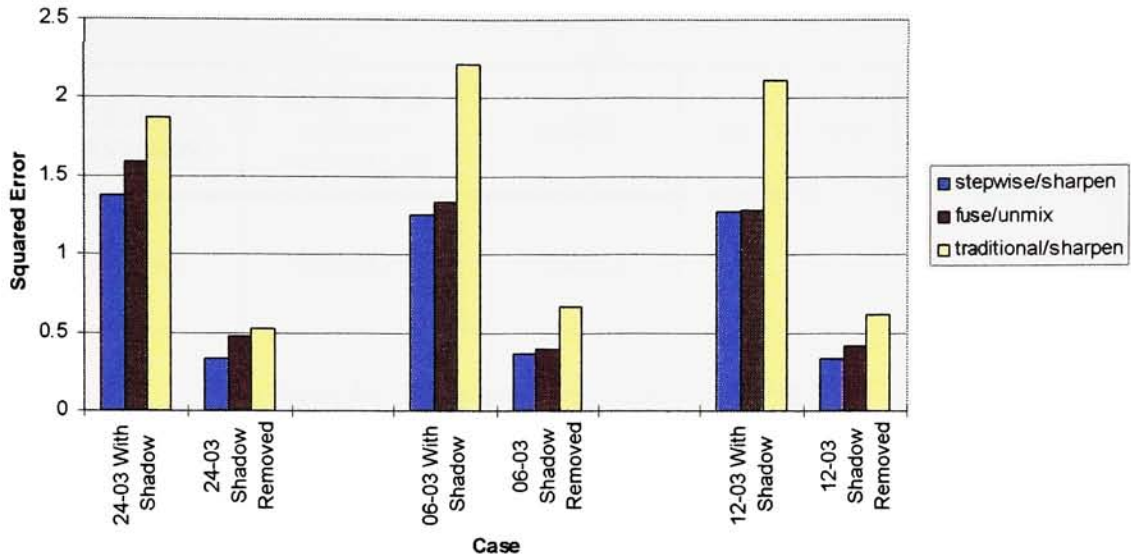


Figure 55: Error Calculated from Shadow Class Map vs Error Calculated from Class Map Without Shadow (two of two)

4.7 Visual Evaluation

Quantitative data was not available when processing the real images. The results of the image enhancement were examined visually, and a qualitative assessment was performed. Visually, the fused/unmixed fraction maps look better than those obtained via unmix/sharpen methods. As shown in Figures 57 through 59, the fuse/unmix fraction maps contain more high frequency data, and have a less blocky appearance than the unmix/sharpen fraction maps, and are preferred to those obtained via the unmix/sharpen method. The fraction maps shown are for the 24-04 case for the DAEDALUS scene. The key to the fraction maps is shown in Figure 56.

DESERT PAVEMENT	DISTURBED DESERT PAVEMENT	ROAD	SILTY SAND	VEGETATION
METAL	SHADOW	CANVAS		

Figure 56: Key to DAEDALUS Scene Fraction Maps

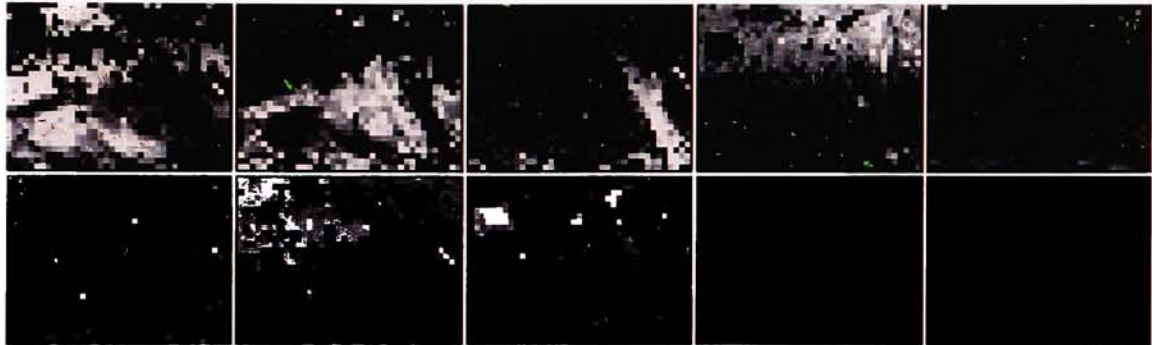


Figure 57: Fraction Maps for DAEDALUS Image (Stepwise Unmix/Sharpen)

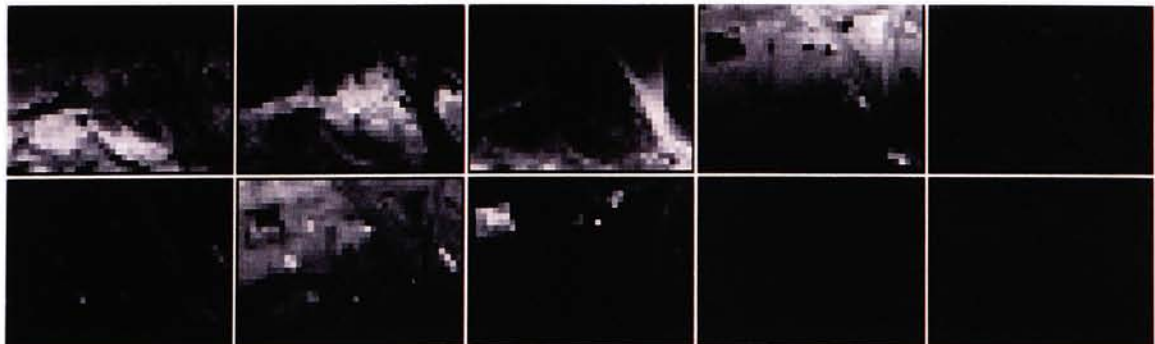


Figure 58: Fraction Maps for DAEDALUS Image (Traditional Unmix/Sharpen)

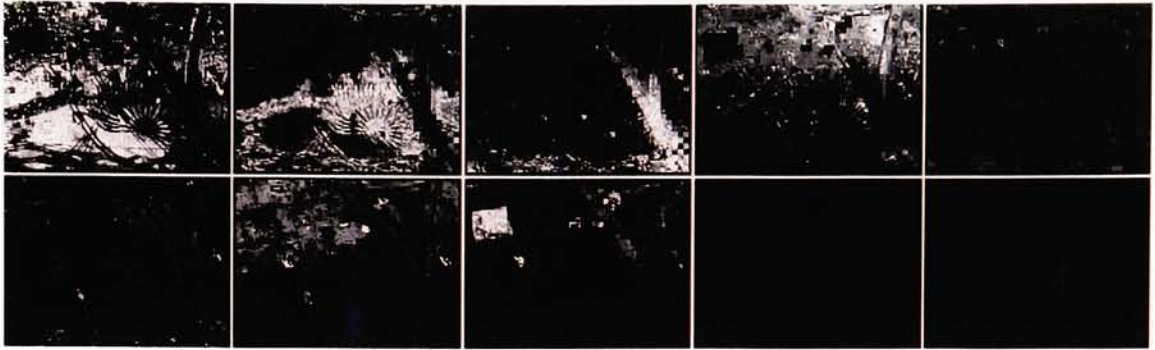


Figure 59: Fraction Maps for DAEDALUS Image (Fuse/Unmix)

The fraction maps generated by the traditional unmix/sharpen method are softer than those generated by the stepwise unmix/sharpen method. In addition, these fraction maps contain no fractions greater than 1.05. A few fractions are less than -0.05, but not many. Note that the fraction maps for the stepwise/sharpen method contain many more pixels requiring fractions less than -0.05 and a few requiring fractions greater than 1.05. For comparison, the DAEDALUS image was degraded by 4X and then unmixed via the stepwise method. This was done in an attempt to compare the enhanced fraction maps with “truth” fraction maps. The result is shown in Figure 60.

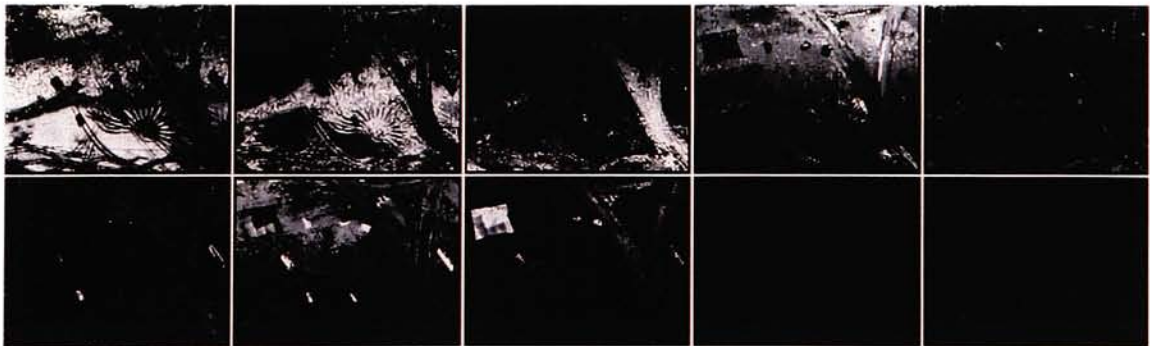


Figure 60: Fraction Maps for DAEDALUS Image (Degraded to 4X)

The fuse/unmix fraction maps look better than the unmix/sharpen maps for every image set. As an example for an urban scene, Rochester fraction maps are shown in Figures 62 through 64. This data represents

enhancing the image from 24X to 4X. The key to the fraction maps is shown in Figure 61. For comparison, truth maps are shown in Figure 65.

GRASS	WATER	ROOF GRAVEL	LOAM	TREES
NEW ASPHALT	OLD ASPHALT	CONCRETE	GLASS	SHINGLES
BRICK	BLUE WOOD	RED WOOD	GREEN WOOD	LIGHT GRAY WOOD

Figure 61: Key to Rochester Scene Fraction Maps

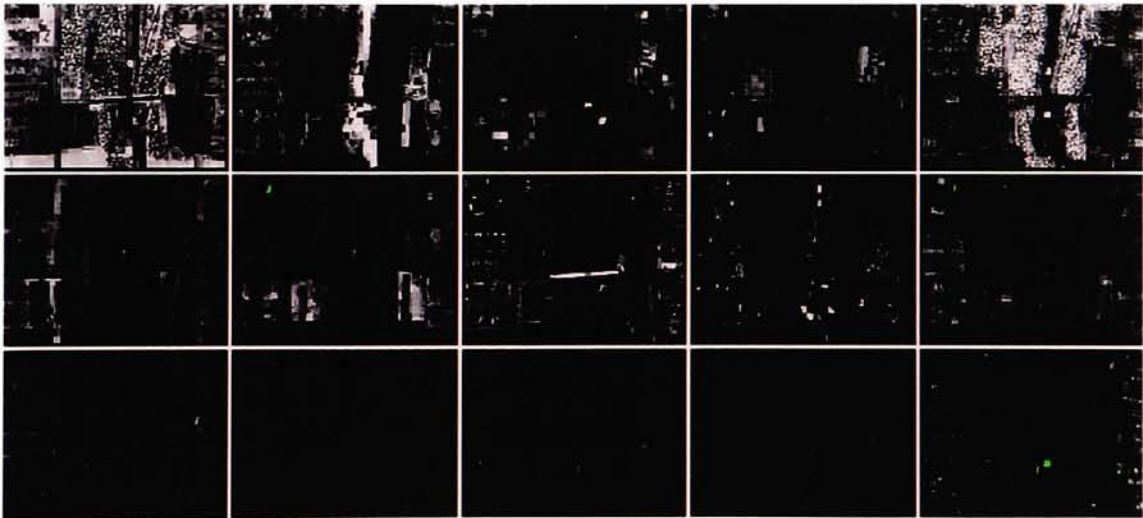


Figure 62: Fraction Maps for Rochester Scene (Fuse/Unmix)

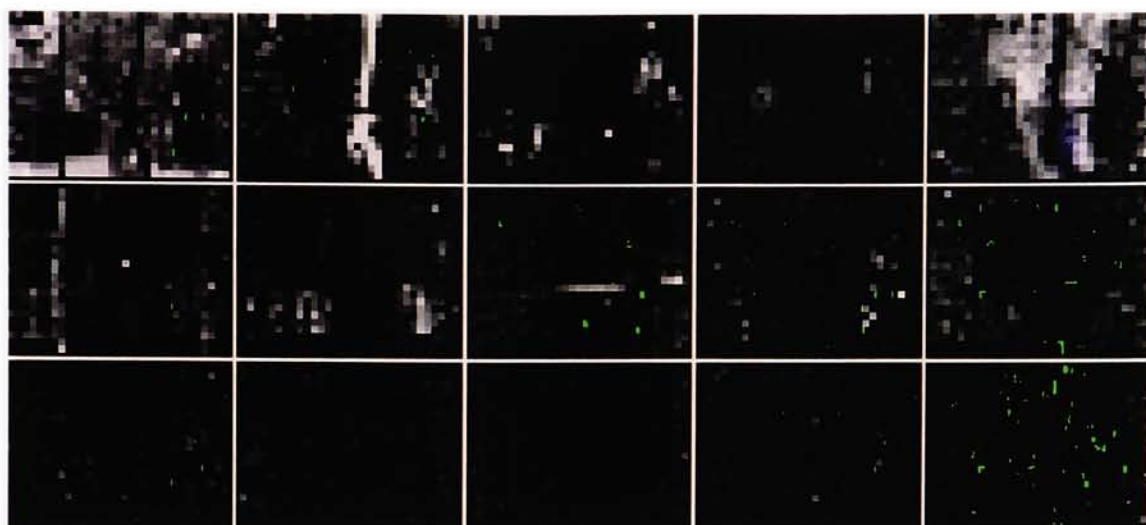


Figure 63: Fraction Maps for Rochester Scene (Stepwise/Sharpen)

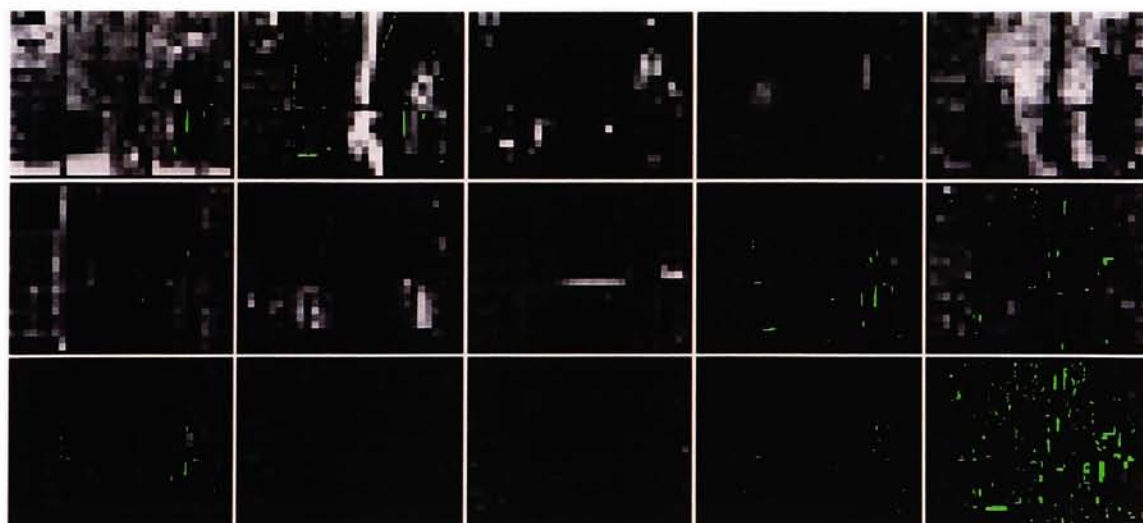


Figure 64: Fraction Maps for Rochester Scene (Traditional/Sharpen)



Figure 65: Rochester Scene Truth Fraction Maps (4X)

Several resolutions were not employed for the HYDICE image. Only one resolution was used, and sharpening was performed with multiple sharpening bands; fusion was performed using only one sharpening band. The HYDICE image was degraded by 16X and fused using a VIS panchromatic image at 2X resolution. The 16X image was unmixed and the fraction maps were sharpened using different combinations of the VIS, NIR, and SWIR sharpening bands. See Figures 67 through 70 for the fraction maps. Use of multiple sharpening bands improves the results. As the number of sharpening bands increases, the sharpened fraction maps look better, with the best sharpened image obtained by using all three sharpening bands. The fraction maps where only sharpening band was used contain many pixels requiring “out of bounds” fractions. There are very few out of bounds pixels in the fraction maps generated using three sharpening bands. The key to the fraction maps is shown in Figure 66.

DESERT PAVEMENT	DISTURBED DESERT PAVEMENT	VEGETATION	ROAD	METAL
CANVAS	SHADOW	SILTY SAND		

Figure 66: Key to HYDICE Scene Fraction Maps

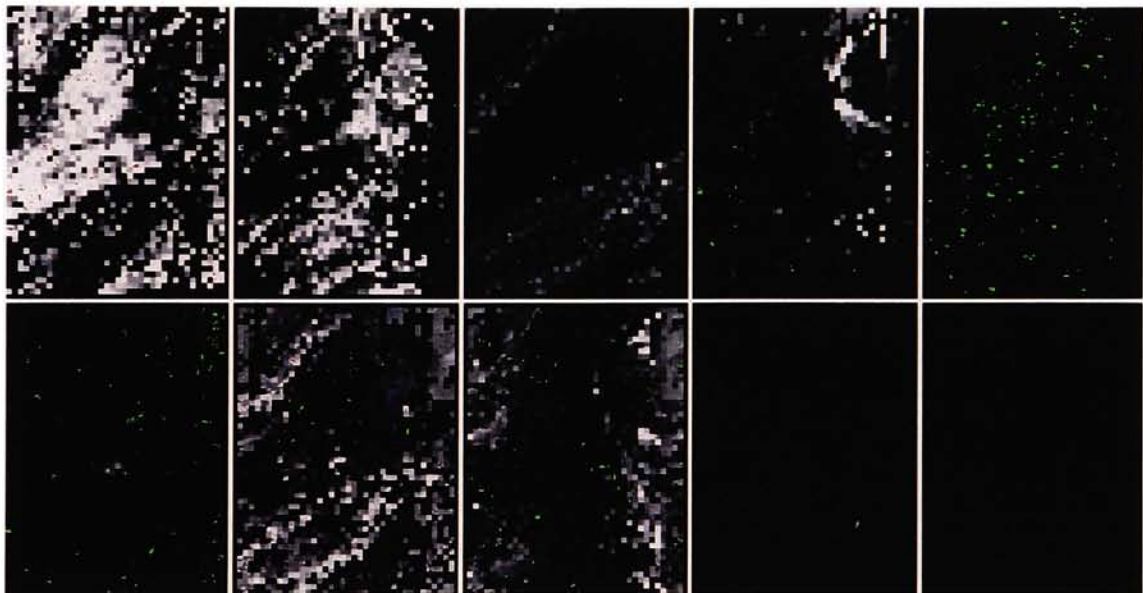


Figure 67: Fraction Maps for HYDICE Image (Stepwise/Sharpen) Using One Sharpening Band

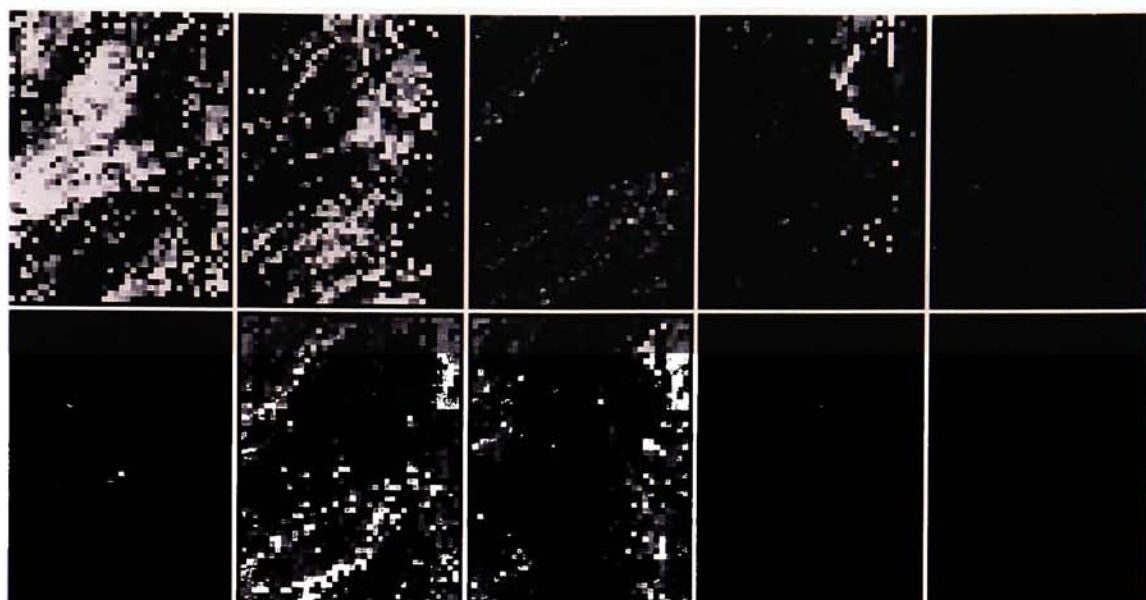


Figure 68: Fraction Maps for HYDICE Image (Stepwise/Sharpen) Using
Three Sharpening Bands

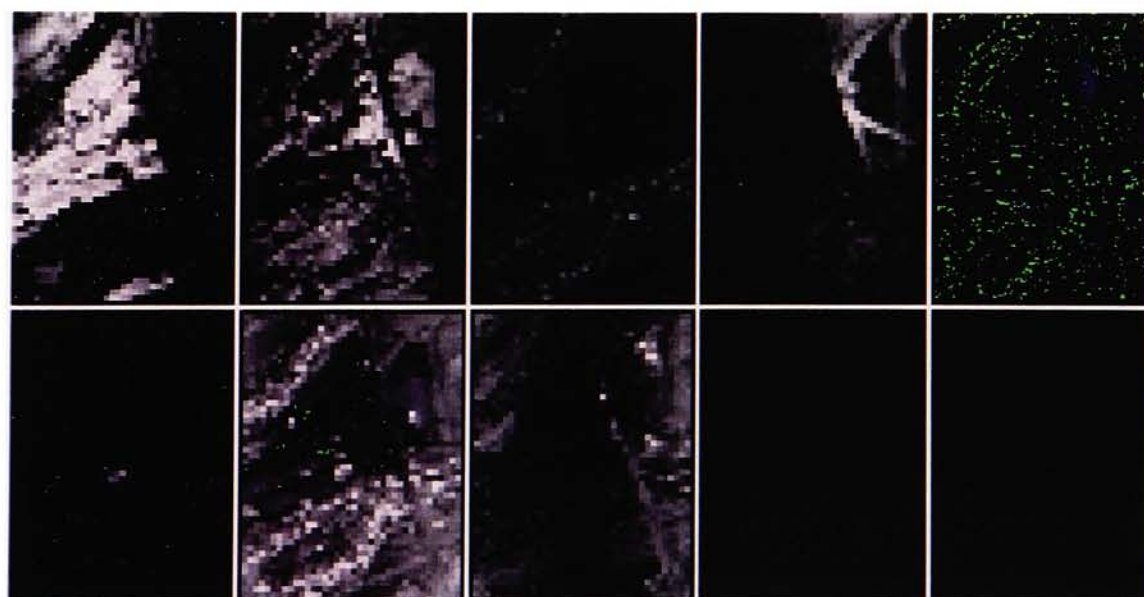
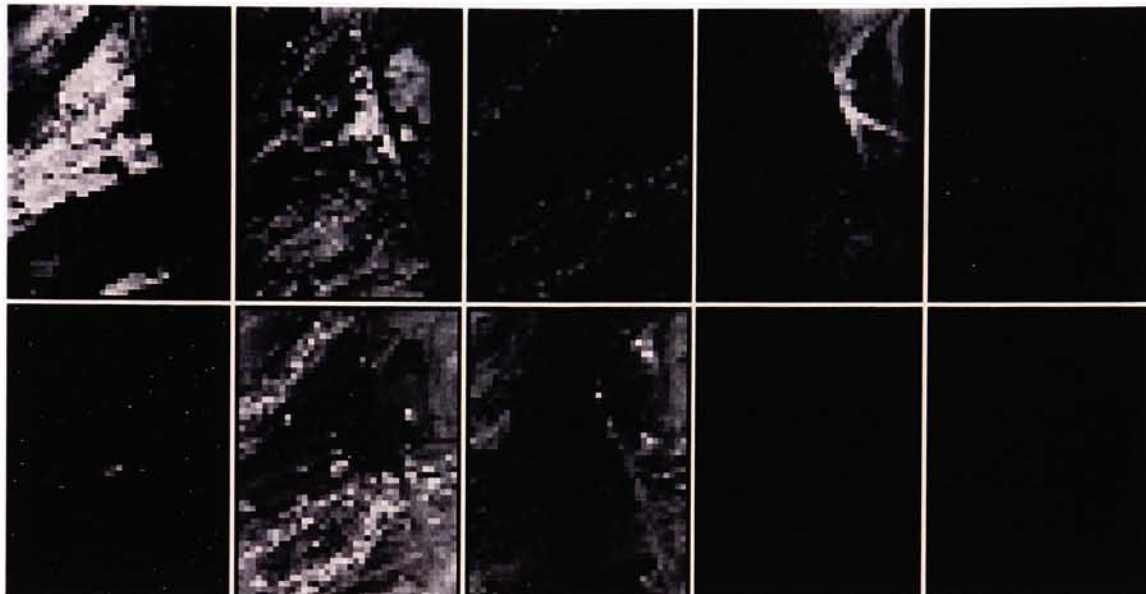


Figure 69: Fraction Maps for HYDICE Image (Traditional/Sharpen) Using
One Sharpening Band



**Figure 70: Fraction Maps for HYDICE Image (Traditional/Sharpen) Using
Three Sharpening Bands**

Similar to other image enhancement scenarios, the Fuse/Unmix fraction contain more useful high-frequency information than the unmix/sharpen fraction maps. The fuse/unmix fraction map follows in Figure 71. For comparison purposes, the original HYDICE scene was degraded to 2X to produce a “truth” image. The results for stepwise unmixing performed on this scene are shown in Figure 72.

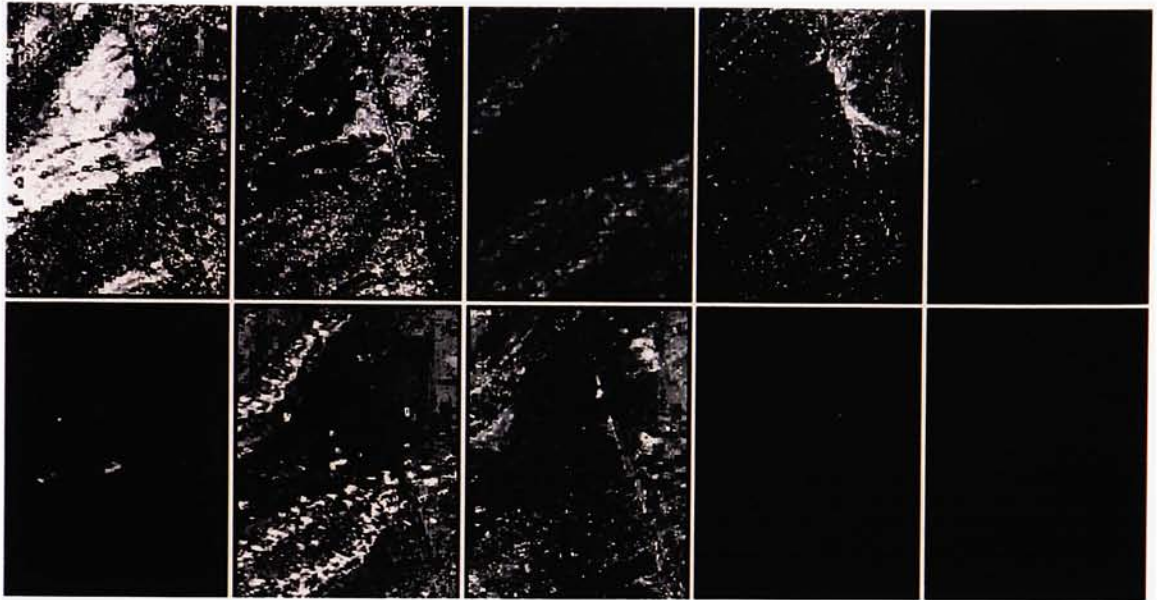


Figure 71: Fraction Maps for HYDICE Image (Fuse/Unmix)

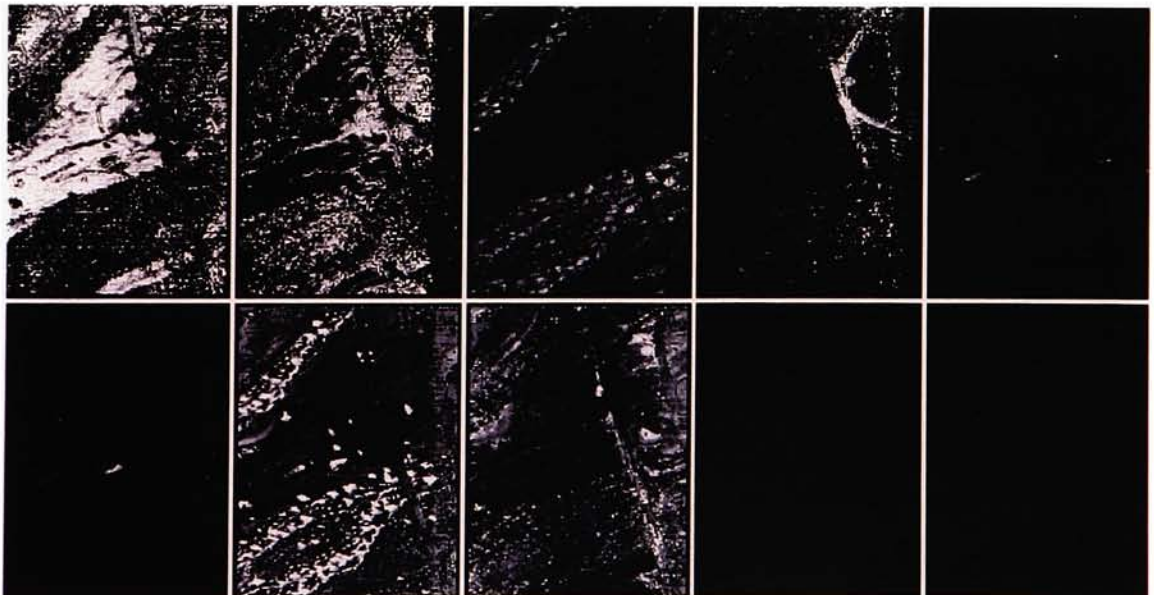


Figure 72: Fraction Maps for HYDICE Image (Degraded to 2X)

5. CONCLUSIONS AND RECOMMENDATIONS

The purpose of this research was to compare the results produced by an unmix/sharpen process with those produced by a fuse/unmix process. In addition, a determination was to be made of which unmixing method (stepwise vs traditional) was the superior method.

5.1 Conclusions Based on Quantitative Data (Truth)

The strongest conclusion reached from the data is that fusing prior to unmixing produces fraction maps with more squared error than unmixing first and then sharpening. Many trials were performed with three different synthetic images (A simple forest scene without texture, which was used to optimize the operation of the unmixing algorithms; and a complex forest scene with texture and shadow, and the Rochester scene which served as the two final DIRSIG images for use in this research.). In over ninety percent of the trials, the fuse/unmix process produced fraction maps with greater squared error than the stepwise unmix/sharpen process.

In the forest scene, the fuse/unmix fraction maps always had higher squared error than the stepwise/sharpen fraction maps. The same was true for the Rochester scene, however, strong conclusions should not be drawn based on this image. Recall that the Rochester scene looked synthetic, whereas the forest scene looked much more realistic, including such features as vignetting. The traditional unmix/sharpen process produced less error than the stepwise/sharpen process. This is a completely unexpected result, and may indicate that there is some artifact in the DIRSIG process of generating an urban scene that incorrectly models real-world processes. The unexpected results may also be due to the fact that the image was generated with 32 materials, which were then grouped (based on similar spectral curves) into fifteen materials. A trial could be performed using a real urban scene, but that would not really answer the question of whether DIRSIG is creating some artifacts in its modeling of urban scenes, because the differences in error values for the two methods are very small, and may not show up with a simple visual examination performed on the real scene (truth is generally not

available when using real scenes). Further study should be performed using high-fidelity urban scenes generated by DIRSIG.

5.2 Conclusions Based on Qualitative Data (Visual)

When success is determined by visual examination, the fuse/unmix process will generate visually superior fraction maps. These fraction maps contain more high frequency data, and generally have a less blocky appearance than the unmix/sharpen fraction maps. The soft and blocky artifacts of the global regression fusion method are often apparent in the fraction maps, but this is acceptable given the better high-frequency content.

Using the unmix/sharpen process, the sharpened material maps are much better than the unsharpened unmixed material maps. In addition, sharpening is better than simply pixel-replicating the low-resolution material maps to achieve maps the same size as the sharpened results. This was demonstrated by Gross (1996). In some cases, the sharpened traditionally-unmixed maps may look better visually than the sharpened stepwise-unmixed maps. However, when the squared error is calculated, the stepwise-unmixed fraction maps always generate less squared error (except for the Rochester scene, and these differences may actually be caused by some DIRSIG artifacts as previously discussed).

So the question of which method is better depends on why the user is performing the unmixing. If overall accuracy is desired, then the unmix/sharpen method is the definite choice. If applications requiring visually enhanced fraction maps are desired, then the fuse/unmix process is the obvious choice.

5.3 Proposed Revision to Squared Error Metric

The squared error metric proposed by Gross (1996) is a valid metric, but does not always provide a good measure of the performance of the algorithm. For example, one of the advantages of the stepwise method is that it can be used to unmix an image using a large spectral library, while traditional unmixing is limited by the number of bands in the image. The squared error metric does not take the number of endmembers unmixed into consideration, so does not allow for a complete description of the performance of the algorithms. Unmixing

a 15-band image with 40 endmembers and producing a squared error of 0.20 is better than unmixing the same image with 10 endmembers and producing a squared error of 0.20, but the error metric does not give that indication. It may be advantageous to create a “normalized” squared error by dividing by the number of endmembers as in

$$SE = \frac{1}{N * L} \sum_{pixels} \sum_{materials} (f_{truth} - f_{test})^2 \quad \text{Eq. 79}$$

where N is the number of pixels in the image, L is the number of endmembers in the reference library. This metric may provide a better measure of the performance of the unmixing algorithms, considering the number of endmembers unmixed.

5.4 Recommendations

Assuming that the user wants to obtain fraction maps at a higher resolution than the multi/hyperspectral images, the traditional unmix/sharpen method is not a recommended method of image enhancement. The computer run times become extremely long for sharpening at scale factors higher than 4X. In addition, the squared error for this method is generally higher than the error for stepwise unmixing followed by sharpening and for fusion followed by unmixing. Equivalent or improved results can be obtained using the stepwise/sharpen method or the fuse/unmix method. Traditional unmixing without further image processing (sharpening) is still a viable option, however the main drawback is that the number of endmembers to be unmixed is limited by the number of bands in the image. The squared error will also generally be higher than if the unmixing were performed via the stepwise method.

5.4.1 Algorithm Improvements

A main focus for the unmixing algorithms used for this research was to implement them all in one popular programming language. The fusion code had been implemented in C, and required images to be in the ERDAS® .LAN format. The unmixing and sharpening algorithms were implemented in MATLAB. If the algorithms had been maintained in these languages, some small amount of error could be attributed to transforming images from a format readable by one language to a format read by the other language. Implementing all algorithms in IDL® eliminated this problem. In addition, IDL® is gaining in popularity as an image processing language, and the results here may be repeated by others in the remote sensing and image processing community. Since the main focus was to implement the algorithms in a common language, rather than optimization, there is definitely opportunity to improve the algorithms for fusion, sharpening, and unmixing.

The fusion algorithm was written to use only one sharpening band, whereas the sharpening routine works well with multiple sharpening bands. Since the fusion algorithm is highly dependent on correlation between the low-resolution multispectral and high-resolution sharpening band, use of more sharpening bands, means a higher likelihood that one or more sharpening bands will be well correlated with the low-resolution image. This should be a relatively easy and straightforward change to implement.

Stepwise unmixing is a “stateless” process, with no memory of the results obtained on the previous (adjacent) pixel. If the pixels in an image were scrambled, and unmixed in a different order, the end result would be exactly the same as if they were not scrambled. A good improvement would be to create a stepwise algorithm which remembers the materials obtained in adjacent pixels. For example, if a pixel was determined to consist of an equal mixture of grass and concrete, there is a likely chance that adjacent pixels will contain one or both of these materials. Using these previous answers should result in a noticeable improvement in run times. A slight modification (which would involve longer run times, but possible improvement in accuracy) would be to perform a two-pass stepwise selection. On the first pass, likely materials are assigned to all pixels in an image.

On the second pass, a neighborhood operation is performed, which employs some type of consistency checking within a selected neighborhood. For example, given a 3 x 3 neighborhood of pixels, the algorithm would ensure that the selected materials and their fractions are spatially located in a realistic manner. Consider the example shown in Figure 73, where the target pixel is in the center of the 3 x 3 neighborhood. The true image contains a boundary between grass and concrete at this point, with the pixels above containing grass, and the pixels below containing concrete. Logically, the target pixel should contain grass and/or concrete. If the algorithm guesses that the pixel contains a large amount of glass, (for example), the glass endmember should be eliminated because, based on surrounding pixels, the likelihood of a large amount of glass in the target pixel is rather low. For scenes with low to medium-frequency content, such a logical check should greatly improve the accuracy of the unmixing algorithm.

Grass: 1.0	Grass 1.0	Grass 1.0
Grass 0.60 Concrete 0.40	Grass 0.60 Concrete 0.40	Grass 0.60 Concrete 0.40
Concrete 1.0	Concrete 1.0	Concrete 1.0

Figure 73: Sample Material Fractions in 3x3 pixel Neighborhood

Improvements in the area of image fusion will not come completely from improving the algorithms developed for this research. Investigation into other areas related to this work may yield interesting answers also.

As stated previously, the squared error metric can be improved. Modifications to the squared error should be investigated. Some metric should be developed which provides an evaluation of the accuracy of the algorithm, considers the size of the library, and considers whether materials are spatially located in a logical manner. Another disadvantage to the squared error metric is that, while it provides an adequate measure of

accuracy of the fraction maps, it provides no indication of the true visual quality. Recall that the fused/unmixed maps contain more high-frequency information, although they typically have higher squared error than the stepwise-unmixed/fused maps. Visually, there is not much difference between fraction maps with a squared error of 0.20 and 0.30 (unless one was generated by the fuse/unmix method and the other generated by the unmix/sharpen method). So the squared error may not give the user all the desired information in one number. Perhaps more than one metric should be used: one for accuracy, spatial location aspects of the fraction maps, and the other for visual quality of the maps.

Unmixing requires a “smart” user to choose adequate endmembers, F-to-enter/exit thresholds, etc. Another goal of further work should be to reduce the need for a smart user. Little investigation was done into the statistics of the library endmembers. There may be some quick mathematical analysis performed on a candidate spectra to determine if it is an adequate endmember for use in unmixing. Endmembers should be spectrally distinct to ensure good stepwise unmixing, and a quick test which determines if an endmember is “sufficiently distinct” can reduce much of the iterative work involved in generating a good spectral library for unmixing. Another related area of investigation is to determine how “similar” spectral curves can be to still accomplish accurate unmixing. This would definitely involve hyperspectral data sets. For example, the curves for disturbed desert pavement and road in the HYDICE data set are very similar, but there are a few characteristic features (“bumps” or “dips” in the spectral curves) which allowed the algorithm to unmix the images with a (visually) high degree of accuracy. A knowledge of how strong a “bump” or “dip” in the spectral curves of the spectral library must be will also assist the user in selecting good endmembers for the unmixing library.

Correlation was examined when dealing with the image fusion algorithms. An interesting question would be to investigate the effect the correlation between the LRXS and HRP has in sharpening the low-resolution fraction maps. The probable answer is that the distinctness of materials in the sharpening library is

more important than correlation, but the possible effects of correlation in sharpening may provide some interesting results.

APPENDIX A: SOLVING THE LSE PROBLEM

Equality constraints reduce the number of free variables in the solution space. The Least Squares Equality (LSE) problem can be solved using an orthogonal transformation presented by Lawson & Hanson (1974).

The function to be minimized is $\|y - Ax\|$ subject to $Cx = d$. Assuming $Cx = d$ is consistent and $n > p = \text{rank}(C)$, then an orthogonal decomposition may be used to partition C into a $p \times p$ submatrix C_1 and a $p \times (n-p)$ zero matrix

$$CV = C \begin{bmatrix} \underbrace{V_1}_{n \times p} & \vdots & \underbrace{V_2}_{n \times (n-p)} \end{bmatrix} = \begin{bmatrix} C_1 & \vdots & 0 \end{bmatrix} \quad \text{Eq A- 1}$$

where V is a $n \times n$ matrix and when multiplied times C , partitions C into C_1 and a zero submatrix. A V matrix may be obtained by a singular value decomposition of C

$$C = USV' \quad \text{Eq A- 2}$$

where S is a $p \times n$ matrix of the singular values of C , and U is a $p \times p$ orthogonal matrix.

The decomposition may be used to solve a p -dimensional subsystem

$$w_1 = C_1^{-1}d \quad \text{Eq A- 3}$$

where w_1 is a p -vector and C_1 can be inverted due to the decomposition.

V is used to decompose the least square system into the same coordinates by

$$AV = A \begin{bmatrix} \underbrace{V_1}_{n \times p} & \vdots & \underbrace{V_2}_{n \times (n-p)} \end{bmatrix} = \begin{bmatrix} \underbrace{A_1}_{m \times p} & \vdots & \underbrace{A_2}_{m \times (n-p)} \end{bmatrix} \quad \text{Eq A- 4}$$

The least square problem is solved by first removing the effect of the transformed constraints from the measured values

$$\tilde{y} = y - A_1 w_1 \quad \text{Eq A- 5}$$

A lower dimensional least square problem (minimize $\|A_2 w_2 - \tilde{y}\|$) is solved for the remaining (n-p) variables (w_2)

$$\tilde{y} = A_2 w_2 \quad \text{Eq A- 6}$$

$$w_2 = (A_2' A_2)^{-1} A_2' \tilde{y} \quad \text{Eq A- 7}$$

The solution is transformed to the original coordinate system to obtain the solution of the original problem by

$$x = Vw = [V_1 : V_2] \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}. \quad \text{Eq A- 8}$$

APPENDIX B: SOLVING THE LSI PROBLEM

While equality constraints reduce the number of free variables in the least squares problem, inequality constraints establish boundaries within the solution space. An iterative solution is required to identify *active* constraints and restrict those affected variables. On each iteration, the active constraints are treated as equality constraints to derive a minimum. The solution to the Least Squares Inequality (LSI) problem is described below.

Minimize $\|Ax-y\|$ subject to $Gx \geq h$ where G is an $r \times n$ matrix, and d is an r -vector. The problem can be divided into two special cases. The Non-Negative Least Squares (NNLS) is a least squares problem requiring all coefficients to be positive,

$$\text{NNLS: Minimize } \|Ax-y\| \text{ subject to } x \geq 0 \quad \text{Eq B- 1}$$

and the Least Distance Programming (LDP) problem is a minimization with respect to the origin.

$$\text{LDP: Minimize } \|x\| \text{ subject to } Gx \geq h \quad \text{Eq B- 2}$$

Any LSI problem can be converted to a LDP problem using an appropriate coordinate transformation. A LDP problem may be solved with a NNLS algorithm.

The heart of solving the LSI problem is the NNLS algorithm. The NNLS routine is used for a straightforward solution to the LDP problem. A NNLS algorithm is presented by Lawson & Hanson (1974). This algorithm consists of two loops. The outer loop brings variables in one at a time. The variable which would have the most positive coefficient is chosen. The loop repeats if the other coefficients remain positive, until all variables are included. If one of the coefficients becomes negative, the inner loop starts. This loop adjusts the step direction to keep the coefficients non-negative. Every time the inner loop is implemented, one of the coefficients is driven to zero. Therefore, a finite number of iterations of the inner loop will be required.

Since there are a finite number of variable combinations considered by the outer loop, the NNLS algorithm must converge.

The LSI problem is solved in the following way:

1. The constraints are decomposed and the dimensionality is reduced.
2. The remaining problem is changed to a LDP problem.
3. The LDP problem is changed to a NNLS problem.
4. The NNLS problem is solved and the solution is transformed back to the original variables.

A LSI problem (minimize $\|Ax-y\|$ subject to $Gx \geq h$) is converted to a LDP problem by the orthogonal decomposition of the matrix A as in

$$A = USV' = \begin{bmatrix} \underbrace{U_1}_{k} : \underbrace{U_2}_{m_2-k} \end{bmatrix} \begin{bmatrix} S & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1' \\ V_2' \end{bmatrix} \quad \text{Eq B- 3}$$

where A is a $m_2 \times n$ matrix of rank k, U is $m_2 \times m_2$ orthogonal, V is $n \times n$ orthogonal, and S is $k \times k$. If single value decomposition is used, S is a diagonal matrix containing the singular values of A. Through a change of variables, where

$$\begin{aligned} x &= V_1 y \\ z &= Sy - U_1' y \\ \tilde{G} &= GV_1 S^{-1} \\ \tilde{h} &= h - \tilde{G}U_1' y \end{aligned} \quad \text{Eq B- 4}$$

$$\text{Problem LSI: minimize } \|Ax-y\| \text{ subject to } Gx \geq h \quad \text{Eq B- 5}$$

is converted to

$$\text{Problem LDP: minimize } \|z\| \text{ subject to } \tilde{G}z \geq \tilde{h} \quad \text{Eq B- 6}$$

The least square problem is solved for z . The original coordinates are obtained by

$$x = V_1 S^{-1} (z + U_1' y) \quad \text{Eq B- 7}$$

A LDP problem is solved by forming a NNLS problem from the constraints. Define the $(n+1) \times m$ matrix A and the $(n+1)$ -vector y as

$$A = \begin{bmatrix} G' \\ h' \end{bmatrix}$$

$$y = \begin{bmatrix} 0 \\ \underbrace{n} \\ 1 \end{bmatrix} \quad \text{Eq B- 8}$$

Now solve the NNLS problem

Problem NNLS: minimize $\|Ax-y\|$ subject to $x \geq 0$.

Where the columns of A define boundary lines within the feasible solution space. The least square problem is solved for x , and the solution is the point in the feasible space with the minimum Euclidian distance to the origin.

The original coordinates are obtained by

$$r = AU - y \quad \text{Eq B- 9}$$

where U is the answer returned by the NNLS algorithm, and r is a $(n+1)$ -vector. Divide r into an n -vector r_1 , and one extra term, r_{n+1}

$$r = \begin{bmatrix} \underbrace{r_1}_n \\ r_{n+1} \end{bmatrix}$$

$$x = \frac{-1}{r_{n+1}} * r_1 \quad \text{Eq B- 10}$$

If the LSI problem has added equality constraints, then the equality constraints can be eliminated (using methods in Appendix A) with a corresponding decrease in the number of variables. Use a change of variables

$$\begin{bmatrix} C \\ A \\ G \end{bmatrix} V = \begin{bmatrix} C_1 & 0 \\ A_1 & A_2 \\ \underbrace{G_1}_{m_1} & \underbrace{G_2}_{n-m_1} \end{bmatrix} \quad \text{Eq B- 11}$$

where $\text{rank}(C) = m_1 < n$. The vector w_1 is found by

$$w_1 = C_1^{-1} d \quad \text{Eq B- 12}$$

where w_1 is a p -vector and C_1 can be inverted due to the decomposition. With the $(n-p)$ -vector w_2 , the original problem is converted to

$$\text{Problem LSI: Minimize } \|A_2 w_2 - (y - A_1 w_1)\| \text{ subject to } G_2 w_2 \geq h - G_1 w_1 \quad \text{Eq B- 13}$$

After solving for w_2 , the solution to the original problem is found by

$$x = Vw = [V_1 : V_2] \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \quad \text{Eq B- 14}$$

To summarize: The LSI problem is solved by the following steps

1. The constraints are decomposed and the dimensionality is reduced (Eq B- 11 , B- 12, and B- 13).
2. The remaining problem is changed to a LDP problem (Eq B- 4 and B- 6).
3. The LDP problem is changed to a NNLS problem (Eq B- 8).
4. The NNLS problem is solved and the solution is transformed back to the original variables (The NNLS answer is transformed by Eq B- 10 to complete the LDP routine. This result is transformed by equation B- 7, and – if equality constraints were included – the final answer is obtained by equation B- 14).

Although the solution to the LSI problem seems awkward, it can be coded as subroutines on a computer in a rather straightforward manner.

APPENDIX C: DATA SETS

The use of synthetic imagery allowed quantitative data to be gathered for the image enhancement trials.

The raw data used to derive the charts in the Results section follow.

CASE	FUSE/UNMIX		STEPWISE/SHARPEN		TRADITIONAL/SHARPEN	
2X (24:12)	FUSE (RMS)	3.02	UNMIX (Squared Err)	0.2106	UNMIX (Squared Err)	0.3628
	FUSE (Edge RMS)	3.15	Time (H:M:S)	0:02:56	Time (H:M:S)	0:04:53
	Time (H:M:S)	0:00:33				
	UNMIX (Squared Err)	0.3195	SHARPEN (Squared Err)	0.2803	SHARPEN (Squared Err)	0.4331
	Time (H:M:S)	0:03:03	Time (H:M:S)	0:01:07	Time (H:M:S)	0:11:53
4X(24:06)	FUSE (RMS)	4.06	UNMIX (Squared Err)	0.2106	UNMIX (Squared Err)	0.3628
	FUSE (Edge RMS)	4.24	Time (H:M:S)	0:02:56	Time (H:M:S)	0:04:53
	Time (H:M:S)	0:00:46				
	UNMIX (Squared Err)	0.3963	SHARPEN (Squared Err)	0.3471	SHARPEN (Squared Err)	0.4995
	Time (H:M:S)	0:41:06	Time (H:M:S)	0:14:14	Time (H:M:S)	2:03:32
6X(24:04)	FUSE (RMS)	4.55	UNMIX (Squared Err)	0.2106	UNMIX (Squared Err)	0.3628
	FUSE (Edge RMS)	4.77	Time (H:M:S)	0:02:56	Time (H:M:S)	0:04:53
	Time (H:M:S)	0:01:44				
	UNMIX (Squared Err)	0.4476	SHARPEN (Squared Err)	0.3888	SHARPEN (Squared Err)	0.5407
	Time (H:M:S)	1:28:19	Time (H:M:S)	1:38:22	Time (H:M:S)	70:27:01
8X(24:03)	FUSE (RMS)	4.94	UNMIX (Squared Err)	0.2106	UNMIX (Squared Err)	0.3628
	FUSE (Edge RMS)	5.22	Time (H:M:S)	0:02:56	Time (H:M:S)	0:04:56
	Time (H:M:S)	0:01:30				
	UNMIX (Squared Err)	1.5859	SHARPEN (Squared Err)	1.3720	SHARPEN (Squared Err)	1.8694
	Time (H:M:S)	2:04:18	Time (H:M:S)	59:29:05	Time (H:M:S)	402:48:16
4x(12:03)	FUSE (RMS)	3.08	UNMIX (Squared Err)	0.2477	UNMIX (Squared Err)	0.5070
	FUSE (Edge RMS)	3.34	Time (H:M:S)	0:13:55	Time (H:M:S)	0:27:49
	Time (H:M:S)	0:01:59				
	UNMIX (Squared Err)	1.2894	SHARPEN (Squared Err)	1.2720	SHARPEN (Squared Err)	2.1187
	Time (H:M:S)	1:36:46	Time (H:M:S)	0:52:30	Time (H:M:S)	12:00:41
2X(06:03)	FUSE (RMS)	2.61	UNMIX (Squared Err)	0.3086	UNMIX (Squared Err)	0.6036
	FUSE (Edge RMS)	2.86	Time (H:M:S)	0:56:39	Time (H:M:S)	1:10:15
	Time (H:M:S)	0:02:15				
	UNMIX (Squared Err)	1.3311	SHARPEN (Squared Err)	1.2536	SHARPEN (Squared Err)	2.2204
	Time (H:M:S)	1:46:34	Time (H:M:S)	0:03:26	Time (H:M:S)	3:39:23

Table 16: Data for Forest Scene With Shadow Endmember (Uncorrected for Shadow)

CASE	FUSE/UNMIX		STEPWISE/SHARPEN		TRADITIONAL/SHARPEN	
2X (24:12)	FUSE (RMS)	3.02	UNMIX (Squared Err)	0.1591	UNMIX (Squared Err)	0.3470
	FUSE (Edge RMS)	3.15	Time (H:M:S)	0:02:56	Time (H:M:S)	0:04:53
	Time (H:M:S)	0:00:33				
	UNMIX (Squared Err)	0.2932	SHARPEN (Squared Err)	0.2323	SHARPEN (Squared Err)	0.4239
	Time (H:M:S)	0:03:03	Time (H:M:S)	0:01:07	Time (H:M:S)	0:11:53
4X(24:06)	FUSE (RMS)	4.06	UNMIX (Squared Err)	0.1591	UNMIX (Squared Err)	0.3470
	FUSE (Edge RMS)	4.24	Time (H:M:S)	0:02:56	Time (H:M:S)	0:04:53
	Time (H:M:S)	0:00:46				
	UNMIX (Squared Err)	0.3861	SHARPEN (Squared Err)	0.2895	SHARPEN (Squared Err)	0.4789
	Time (H:M:S)	0:41:06	Time (H:M:S)	0:14:14	Time (H:M:S)	2:03:32
6X(24:04)	FUSE (RMS)	4.55	UNMIX (Squared Err)	0.1591	UNMIX (Squared Err)	0.3470
	FUSE (Edge RMS)	4.77	Time (H:M:S)	0:02:56	Time (H:M:S)	0:04:53
	Time (H:M:S)	0:01:44				
	UNMIX (Squared Err)	0.4370	SHARPEN (Squared Err)	0.3177	SHARPEN (Squared Err)	0.5055
	Time (H:M:S)	1:28:19	Time (H:M:S)	1:38:22	Time (H:M:S)	70:27:01
8X(24:03)	FUSE (RMS)	4.94	UNMIX (Squared Err)	0.1591	UNMIX (Squared Err)	0.3470
	FUSE (Edge RMS)	5.22	Time (H:M:S)	0:02:56	Time (H:M:S)	0:04:56
	Time (H:M:S)	0:01:30				
	UNMIX (Squared Err)	0.4778	SHARPEN (Squared Err)	0.3363	SHARPEN (Squared Err)	0.5243
	Time (H:M:S)	2:04:18	Time (H:M:S)	59:29:05	Time (H:M:S)	402:48:16
4x(12:03)	FUSE (RMS)	3.08	UNMIX (Squared Err)	0.2317	UNMIX (Squared Err)	0.5131
	FUSE (Edge RMS)	3.34	Time (H:M:S)	0:13:55	Time (H:M:S)	0:27:49
	Time (H:M:S)	0:01:59				
	UNMIX (Squared Err)	0.4151	SHARPEN (Squared Err)	0.3385	SHARPEN (Squared Err)	0.6203
	Time (H:M:S)	1:36:46	Time (H:M:S)	0:52:30	Time (H:M:S)	12:00:41
2X(06:03)	FUSE (RMS)	2.61	UNMIX (Squared Err)	0.3132	UNMIX (Squared Err)	0.6129
	FUSE (Edge RMS)	2.86	Time (H:M:S)	0:56:39	Time (H:M:S)	1:10:15
	Time (H:M:S)	0:02:15				
	UNMIX (Squared Err)	0.3936	SHARPEN (Squared Err)	0.3637	SHARPEN (Squared Err)	0.6658
	Time (H:M:S)	1:46:34	Time (H:M:S)	0:03:26	Time (H:M:S)	3:39:23

Table 17: Data for Forest Scene With Shadow Endmember (Corrected for Shadow)

CASE	FUSE/UNMIX		STEPWISE/SHARPEN		TRADITIONAL/SHARPEN	
2X (24:12)	FUSE (RMS)	3.02	UNMIX (Squared Err)	0.1925	UNMIX (Squared Err)	0.3476
	FUSE (Edge RMS)	3.15	Time (H:M:S)	0:15:41	Time (H:M:S)	0:02:32
	Time (H:M:S)	0:00:33				
	UNMIX (Squared Err)	0.3251	SHARPEN (Squared Err)	0.2657	SHARPEN (Squared Err)	0.4238
	Time (H:M:S)	0:59:11	Time (H:M:S)	0:00:24	Time (H:M:S)	0:06:44
4X(24:06)	FUSE (RMS)	4.06	UNMIX (Squared Err)	0.1925	UNMIX (Squared Err)	0.3476
	FUSE (Edge RMS)	4.24	Time (H:M:S)	0:15:41	Time (H:M:S)	0:02:32
	Time (H:M:S)	0:00:46				
	UNMIX (Squared Err)	0.4238	SHARPEN (Squared Err)	0.3228	SHARPEN (Squared Err)	0.4784
	Time (H:M:S)	2:32:28	Time (H:M:S)	0:11:45	Time (H:M:S)	2:57:54
6X(24:04)	FUSE (RMS)	4.55	UNMIX (Squared Err)	0.1925	UNMIX (Squared Err)	0.3476
	FUSE (Edge RMS)	4.77	Time (H:M:S)	0:15:41	Time (H:M:S)	0:02:32
	Time (H:M:S)	0:01:44				
	UNMIX (Squared Err)	0.4778	SHARPEN (Squared Err)	0.3510	SHARPEN (Squared Err)	0.5058
	Time (H:M:S)	3:48:30	Time (H:M:S)	2:37:48	Time (H:M:S)	19:39:55

Table 18: Data for Forest Scene Without Shadow Endmember

CASE	FUSE/UNMIX		STEPWISE/SHARPEN		TRADITIONAL/SHARPEN	
2X (24:12)	FUSE (RMS)	1.80				
	FUSE (Edge RMS)	2.48	Unmix Time (H:M:S)	0:01:40	Unmix Time (H:M:S)	0:01:13
	Time (H:M:S)	0:00:30				
	Unmix Time (H:M:S)	0:08:47	Sharpen Time (H:M:S)	0:01:10	Sharpen Time (H:M:S)	0:05:11
4X(24:06)	FUSE (RMS)	3.21				
	FUSE (Edge RMS)	5.87	Unmix Time (H:M:S)	0:01:40	Unmix Time (H:M:S)	0:01:13
	Time (H:M:S)	0:01:10				
	Unmix Time (H:M:S)	0:36:10	Sharpen Time (H:M:S)	0:48:19	Sharpen Time (H:M:S)	0:50:53
6X(24:04)	FUSE (RMS)	3.95				
	FUSE (Edge RMS)	8.42	Unmix Time (H:M:S)	0:01:40	Unmix Time (H:M:S)	0:01:13
	Time (H:M:S)	0:02:05				
	Unmix Time (H:M:S)	1:02:05	Sharpen Time (H:M:S)	5:54:36	Sharpen Time (H:M:S)	94:49:03
8X(24:03)	FUSE (RMS)	4.32				
	FUSE (Edge RMS)	10.33	Unmix Time (H:M:S)	0:01:40	Unmix Time (H:M:S)	0:01:13
	Time (H:M:S)	0:05:17				
	Unmix Time (H:M:S)	2:04:18	Sharpen Time (H:M:S)	33:17:04	Sharpen Time (H:M:S)	111:17:52
4x(12:03)	FUSE (RMS)	3.16				
	FUSE (Edge RMS)	8.02	Unmix Time (H:M:S)	0:16:11	Unmix Time (H:M:S)	0:12:11
	Time (H:M:S)	0:05:09				
	Unmix Time (H:M:S)	1:03:19	Sharpen Time (H:M:S)	1:37:58	Sharpen Time (H:M:S)	1:47:37
2X(06:03)	FUSE (RMS)	2.57				
	FUSE (Edge RMS)	5.21	Unmix Time (H:M:S)	2:31:58	Unmix Time (H:M:S)	2:28:49
	Time (H:M:S)	0:07:39				
	Unmix Time (H:M:S)	2:55:06	Sharpen Time (H:M:S)	0:31:25	Sharpen Time (H:M:S)	0:30:42

Table 19: Data for DAEDALUS scene

CASE	FUSE/UNMIX		STEPWISE/SHARPEN		TRADITIONAL/SHARPEN	
2X (24:12)	FUSE (RMS)	15.63	UNMIX (Squared Err)	0.1096	UNMIX (Squared Err)	0.0900
	FUSE (Edge RMS)	16.60	Time (H:M:S)	0:13:30	Time (H:M:S)	0:03:53
	Time (H:M:S)	0:00:08				
	UNMIX (Squared Err)	0.2471	SHARPEN (Squared Err)	0.2250	SHARPEN (Squared Err)	0.2058
4X(24:06)	Time (H:M:S)	1:06:26	Time (H:M:S)	0:05:11	Time (H:M:S)	0:10:06
	FUSE (RMS)	20.77	UNMIX (Squared Err)	0.1096	UNMIX (Squared Err)	0.0900
	FUSE (Edge RMS)	22.69	Time (H:M:S)	0:13:30	Time (H:M:S)	0:03:53
	Time (H:M:S)	0:00:20				
6X(24:04)	UNMIX (Squared Err)	0.3520	SHARPEN (Squared Err)	0.3445	SHARPEN (Squared Err)	0.3246
	Time (H:M:S)	3:47:32	Time (H:M:S)	1:24:57	Time (H:M:S)	3:07:30
	FUSE (RMS)	22.58	UNMIX (Squared Err)	0.1096	UNMIX (Squared Err)	0.0900
	FUSE (Edge RMS)	24.65	Time (H:M:S)	0:13:30	Time (H:M:S)	0:03:53
8X(24:03)	Time (H:M:S)	0:00:42				
	UNMIX (Squared Err)	0.4175	SHARPEN (Squared Err)	0.4112	SHARPEN (Squared Err)	0.3909
	Time (H:M:S)	5:01:08	Time (H:M:S)	36:11:38	Time (H:M:S)	38:57:33
	FUSE (RMS)	24.13	UNMIX (Squared Err)	0.1096	UNMIX (Squared Err)	0.0900
4x(12:03)	FUSE (Edge RMS)	26.24	Time (H:M:S)	0:13:30	Time (H:M:S)	0:03:53
	Time (H:M:S)	0:01:12				
	UNMIX (Squared Err)	1.0384	SHARPEN (Squared Err)	1.0378	SHARPEN (Squared Err)	0.9906
	Time (H:M:S)	11:18:30	Time (H:M:S)	340:42:15	Time (H:M:S)	451:14:37
2X(06:03)	FUSE (RMS)	18.95	UNMIX (Squared Err)	0.1871	UNMIX (Squared Err)	0.1713
	FUSE (Edge RMS)	21.45	Time (H:M:S)	3:25:50	Time (H:M:S)	0:49:37
	Time (H:M:S)	0:01:45				
	UNMIX (Squared Err)	0.6720	SHARPEN (Squared Err)	0.6732	SHARPEN (Squared Err)	0.6371
	Time (H:M:S)	8:43:00	Time (H:M:S)	0:38:26	Time (H:M:S)	1:18:39

Table 20: Data for Rochester Scene

6. REFERENCES

1. Braun, G., Quantitative Evaluation of Six Multi-spectral, Multi-resolution Image Merger Routines, *Master's Thesis, Rochester Institute of Technology*, 1992.
2. Chavez, P.S., G.L. Berlin, and M.A. Tarabzouni, Discriminating Lithologies and Surficial Deposits in the AL Hisma Plateau Region of Saudi Arabia With Digitally Combined Landsat MSS and SIR-A Images: *Proceedings of the National Conference on Resource Management Applications: Energy and Environment*, Vol. 4, San Francisco, CA, pp. 22-34, 1983.
3. Chavez, P.S., S.C. Sides, and J.A. Anderson, Comparison of Three Different Methods to Merge Multiresolution and Multispectral Data: Landsat TM and SPOT Panchromatic: *Photogrammetric Engineering and Remote Sensing*, Vol. 57, No. 3, pp. 295-303, 1991.
4. Clark, R.N., A.J. Gallagher, and G.A. Swayze, Material Absorption Band Depth Mapping of Imaging Spectrometer Data Using a Complete Band Shape Least-Squares Fit With Library Reference Spectra. *Proceedings of the Second Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Workshop*, JPL Publication 90-54, pp. 176-186, 1990.
5. Clark, R.N., G.A. Swayze, A. Gallagher, N. Gorelick, and F. Kruse. Mapping with Imaging Spectrometer Data Using the Complete Band Shape Least-Squares Algorithm Simultaneously Fit to Multiple Spectral Features from Multiple Materials. *Proceedings of the Third Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Workshop*, JPL Publication 91-28, pp. 2-3, 1991.
6. Daily, M., T. Farr, C. Elachi, and G. Schaber, Geologic Interpretation From Composited Radar and Landsat Imagery. *Photogrammetric Engineering and Remote Sensing*, Vol. 45, No. 89, pp. 1109-1116, 1979.
7. Draper, N.R., Smith, H. *Applied Regression Analysis*, 2nd Edition., 1981.
8. Green, A.A., Berman, M., Switzer, P., and Craig, M.D. A Transformation for Ordering Multispectral Data in Terms of Image Quality with Implications for Noise Removal: *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 26, No. 1, pp. 65-74.
9. Green, R.O., Conel, J.E., Roberts, D.A., Estimation of Aerosol Optical Depth, Pressure Elevation, Water Vapor and Calculation of Apparent Surface Reflectance from Radiance Measured by the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Using a Radiative Transfer Code. *Proceedings of SPIE: Imaging Spectrometry of the Terrestrial Environment*, Vol. 1932, 14-15 April 1993, pp. 2-11, 1993.
10. Gross, H.N., An Image Fusion Algorithm for Spatially Enhancing Spectral Mixture Maps, *Doctoral Dissertation, Rochester Institute of Technology*, 1996.

11. Lauer, D.T. and W.J. Todd, Land Cover Mapping with Merged Landsat RBV and MSS Stereoscopic Images. *Proceedings of the ASP Fall Technical Conference*, American Society for Photogrammetry, San Francisco, CA., pp. 680-689, 1981.
12. Lawson, C.L., Hanson, R.J., *Solving Least Squares Problems.*, 1974.
13. MUG. Important Spectral Regions for Representative Applications (Appendix II), *Multispectral Users Guide*, Geodynamics Corp., August, 1995.
14. Munechika, C.K., Merging Panchromatic and Multispectral Images for Enhanced Image Analysis, *Master's Thesis, Rochester Institute of Technology*, 1990.
15. Munechika, C.K., Warnick, J.S., Salvaggio, C., Schott, J.R., Resolution Enhancement of Multispectral Image Data to Improve Classification Accuracy. *Photogrammetric Engineering and Remote Sensing*, Vol. 59, 1 (January), pp. 67-72, 1993.
16. Pradines, D., Improving SPOT Images Size and Multispectral Resolution. *Proceedings of the SPIE: Earth Remote Sensing Using Landsat Thematic Mapper and SPOT Sensor Systems*, Vol. 660, pp. 98-102, 1986.
17. Price, J.C., Combining Panchromatic and Multispectral Imagery from Dual Resolution Satellite Instruments. *Remote Sensing of the Environment*, Vol. 21, pp. 119-128, 1987.
18. Roberts. D.A., Adams, J.B., Smith, M.O., Nonphotosynthetic Vegetation, and Soils in AVIRIS Data, *Remote Sensing of the Environment*, Vol. 44, pp. 255-269, 1993.
19. Schlingmeier, D. Resolution Enhancement of Thermal Infrared Images Via High Resolution Class-Map and Statistical Methods, *Master's Thesis, Rochester Institute of Technology*, 1997.
20. Schowengerdt, R.A., Reconstruction of Multispatial, Multispectral Image Data Using Spatial Frequency Content, *Photogrammetric Engineering and Remote Sensing*, Vol. 46, 10 (October), pp. 1325-1334, 1980
21. Smith, M.O., Ustin, S.L., Adams, J.B., Gillespie, A.R., Vegetation in Deserts I: A Regional Measure of Abundance from Multispectral Images, *Remote Sensing of the Environment*, Vol. 31, pp. 1-26, 1990.
22. Smith, M.O., Ustin, S.L., Adams, J.B., Gillespie, A.R., Vegetation in Deserts II: Environmental Influences on Regional Abundance. *Remote Sensing of Environment*, Vol. 31, pp. 27-52, 1990.
23. Welch, R. and M. Ehlers, Merging Multi-resolution SPOT HRV and Landsat TM Data, *Photogrammetric Engineering and Remote Sensing*, Vol. 53, No. 3, pp. 301-303, 1987.
24. White, R. Digital Image Validation of Rochester Institute of Technology's (RIT's) Digital Image and Remote Sensing Image Generation (DIRSIG) Model - Reflective Region, *Master's Thesis, Rochester Institute of Technology*, 1996.

